CONTATORI Moltiplicatori di tensione python robot wireless inseritore telefonico INTERFACCIA MICROCONTROLLORI Arduino ANALIZZATORE CONSUMI PIC MikroBASIC SystemVerilog GPRS

fare elettronica
www.farelettronica.com
317 NOVEMBRE Mensile Anno XXVII € 6,00

ARDUINO

> Robot Wireless
> Comunicazione GPRS

GREEN
un circuito per scoprire i consumi

Per imparare

ESPERIMENTI CON I CONTATORI DIGITALI

COME COLLEGARSI AD UN MICROCONTROLLORE

UTILIZZARE GLI SHIFT REGISTER

ISSN 1591-2272

CLASSIFICATA N. 1
Puntualità nelle consegne

GO DIGIKEY.IT

Firmware

l'unica rivista italiana nativa digitale per i professionisti dell'elettronica e dedicata, principalmente, ai microcontrollori, dispositivi FPGA, componentistica analogica e approfondimenti sulle tecnologie

- non più solo testo ma anche video!
- possibilità di grande interazione per il lettore
- links a tutte le varie risorse aggiuntive
- possibilità di cercare un testo nella rivista corrente e nell'archivio (!)
- possibilità di stampare tutta la rivista o anche solo alcune parti
- possibilità di leggere la rivista offline scaricandola sul PC
- possibilità di leggere la rivista con gli e-reader (compreso iPhone e iPad)
- moltissime riviste in archivio GRATIS per i nuovi abbonati
- abbonamento a partire da € 19.50
- Possibilità di scaricare la rivista in pdf (solo per gli abbonati)

Richiedi la tua copia omaggio qui:
http://mailing.fwonline.eu

CODICE MIP 2814072

CLASSIFICATA N. 1

Sito Web più visitato

Digi-Key®
CORPORATION

GO DIGIKEY.IT

Classificata come numero uno per i siti Web più visitati dai distributori, dagli ingegneri elettronici del Nord America nello Studio UBM/EE Times 2011 sulle preferenze dei clienti dei distributori (giugno 2011).
Digi-Key è un distributore autorizzato di tutti questi fornitori. Ogni giorno vengono aggiunti nuovi prodotti. © 2011 Digi-Key Corporation, 701 Brooks Ave. South, Thief River Falls, MN 56701, USA

CODICE MIP 2800839

sommario

317 *novembre 2011*

Zoom in

22 UTILIZZARE ARDUINO CON GSM/GPRS

Arduino rappresenta un classico esempio di architettura hardware open source, tanto da essere utilizzato in qualsiasi contesto operativo: dalle applicazioni con Bluetooth al Wi-Fi fino ad arrivare al GSM/GPRS.

di Francesco Pentella

Progettare & costruire

28 INSERITORE MAGNETICO

Un inseritore per antifurto realizzato completamente con materiale di recupero che non utilizza un Pic! Per essere precisi non utilizza alcuna elettronica…

di Daniele Cappa

34 ANALIZZATORE DI RETE ENEL

Se non avete mai pensato di controllare il consumo di energia elettrica del frigo nuovo per capire se le caratteristiche di risparmio energetico tanto decantate dal venditore siano reali, oppure per conoscere il consumo della lavatrice per un determinato lavaggio o ancora quanto costa in termini di energia elettrica scaldare una pizza surgelata, ora potete farlo.

di Silvano Breggion

46 INTERFACCIA D'USCITA PER DATI A 8 BIT

La grande richiesta di modernità e di efficienza ha praticamente ridotto all'osso il prezzo dei processori e dei componenti affini: la moderna tecnologia non ne può più fare a meno ed essi sono ormai presenti in ogni dispositivo; vediamo insieme qualche idea per utilizzarli al meglio.

di Giorgio Ober

L'angolo di Mr A.Keer

54 SHIFT REGISTER

(parte seconda)

La presenza dei Flip-Flops D-Type nell'ambito dei Registri a Scorrimento ribadisce la loro straordinaria utilità; dopo l'analisi dettagliata della modalità SISO vediamo ora quella SIPO, una variante della precedente che offre nuovi interessanti orizzonti applicativi.

di Giorgio Ober

Rispondi e... **VINCI!** *pag.* **45**

DIVERTITI E METTI ALLA PROVA LE TUE CONOSCENZE CON **ELETTRO QUIZ** E VINCI OGNI MESE ESCLUSIVI PREMI!

GLI ARTICOLI CONTRASSEGNATI COL SIMBOLO WWW sono già disponibili in formato PDF* all'indirizzo ***www.farelettronica.com/club***

**Puoi iscriverti al CLUB di Fare Elettronica versando una piccola quota annuale.*

Imparare & approfondire

68 ESPERIMENTI CON I CONTATORI

I contatori sono dispositivi digitali in grado di effettuare il conteggio di impulsi applicati in ingresso. A seconda dei modelli, possono essere sincroni o asincroni, contare in avanti o all'indietro, essere dotati di ingressi di reset e di enable, essere presettabili o meno e utilizzabili in modalità binaria e/o BCD. Semplici esperimenti pratici su bread-board e simulazioni circuitali al calcolatore possono essere di aiuto nell'acquisizione pratica di questi concetti.

di Antonio Giannico

82 GESTIAMO LE STRINGHE CON MIKROBASIC

Da questo numero di "Fare Elettronica" inizia un corso avanzato sulla programmazione dei microcontrollori PIC, a grande richiesta dei lettori. Esso può considerarsi a tutti gli effetti il naturale prosieguo del precedente che aveva riscosso, a suo tempo, un successo strepitoso.

di Giovanni Di Maria

90 CORSO DI PHYTON (parte quinta)

INSERIMENTO DA TASTIERA

I programmi che abbiamo scritto finora sono piuttosto banali, nel senso che non accettano inserimenti di dati da parte dell'operatore, limitandosi a eseguire sempre le stesse operazioni. Python fornisce un insieme di funzioni predefinite che permettono di inserire dati da tastiera.

di Massimiliano Miocchi

94 CORSO SYSTEMVERILOG (parte sesta)

Introduzione alla libreria OVM Concetti di base, interfacce, transazioni e driver.

di Mariano Severi

Robot Zone

102 ARDUINOBOT (parte prima)

UN ROBOT ESPLORATORE CON ARDUINO

Il robot esploratore, cui ho attribuito il nome ArduinoBOT, è basato sulla scheda di controllo Arduino (2009 o UNO). Per muoversi, utilizza due ruote azionate da motoriduttori comandati da una scheda motor shield che utilizza un integrato SN754410.

di Adriano Gandolfo

rubriche

7	Editoriale
10	Idee di progetto
14	News
18	Eventi
44	Info
45	Elettroquiz
108	IESHOP

ABBIAMO BISOGNO DEL TUO AIUTO!

FACCI CONOSCERE LE TUE PREFERENZE VOTANDO IL TUO ARTICOLO PREFERITO SUL SITO WWW.FARELETTRONICA.COM

elenco inserzionisti

Digi-Key Corporation Pag. 1,3
701 Brooks Ave South - 56701 Thief River Falls, MN
Tel. 800 338 4105 x1454 – www.digikey.it

Elettroshop Pag. 63, 93
Via Giotto, 7 - 20032 Cormano (MI)
Tel. 02 66504755 - www.elettroshop.com

Elenos Pag. 92
Via Amendola, 9 - 44028 Poggio Renatico (FE)
Tel. 0532 829965 – www.elenos.com

Evr Pag. 105
Viale Kennedy, 96 - 20027 Rescaldina (MI)
Tel. 0331 1815404 - www.evr-electronics.com

Exposition Service Pag. 18
V.le Dante Alighieri 54 - 48022 Lugo (RA)
Tel. 0545 27548 - www.mondoelettronica.net

Farnell Italia Pag. 9
Corso Europa 20 - 22 - 20020 Lainate (MI)
Tel. 02 93995(1) - www.farnell.com

Futura Elettronica Pag. 6
Via Adige 11 - 21013 Gallarate (VA)
Tel. 0331 799775 - www.futuranet.it

GRIX Pag. 87
www.grix.it

Italfiere Pag. 19
Via Caduti di via Fani 65 - 47023 Cesena (FC)
Tel. 0547 415674 - www.italfiere.net

LeCroy Pag. 13
via E. Mattei Valecenter 1/c/102a - 30020 Marcon (VE)
Tel. 041 5997011 - www.lecroy.com

Micromed Pag. 43
Via Valpadana 126B/2 - 00141 Roma (RM)
Tel. 06/82000066 - www.micromed.it

MikroElektronika Pag. 39
Visegradska, 1A - 11000 Belgrade
Tel. +381 11 3628830 - www.mikroe.com

Millennium Dataware Pag. 14
Corso Repubblica 48 - 15057 Tortona (AL)
Tel. 0131 860254 - www.mdsrl.it

PCB-Pool Pag. 16
Bay 98-99 - Shannon Free Zone
Shannon - County Clare
Tel. 02 64672645 - www.pcb-pool.com

Pordenone Fiere Pag. 17
V.le Treviso 1 - 33170 Pordenone (PN)
Tel. 0434 232111 - www.fierapordenone.it

R.C.C. Pag. 15
Via G. Di Vittorio 19 - 20097 San Donato Milanese (MI)
Tel. 02 51876194 - www.rccitaly.com

RS Components Pag. IVcop
Via M. V. De Vizzi 93/95 - 20092 Cinisello Balsamo (MI)
Tel. 02 66058257 – rswww.it

Tecnoimprese Pag. 99
Via Console Flaminio, 19 - 20134 (MI)
Tel. 02 210.111.1 - www.fortronic.it

Teltools Pag. 107
Via Della Martinella 9 - 20152 Milano (MI)
www.teltools.it

LUPUS IN FABULA

Arduino, 20, 102 - Euro Technologies, 16
Ikalogic, 14 - Image S, 14
Linear Technology, 16
Maxim Integrated Products, 15
Microchip, 34, 82 - Mikroelektronika, 82
Renesas Electronics, 15 - Telit, 20

NUOVE TELECAMERE CON TECNOLOGIA Effio™

Qualità e Prestazioni senza precedenti

TELECAMERA IR VARIFOCAL a COLORI con DSP SONY EFFIO-E 700 LINEE TV

- Sensore da 1/3" Exview CCD;
- Sensibilità 0 lux (IR On);
- 976x582 pixel (PAL);
- Ottica varifocal manuale 2,8-11mm;
- 42 LED IR (portata ~40m);
- Alimentazione 12Vdc;
- Dimensioni 200x17x83mm;
- Da esterno (IP66);
- Completa di staffa.

CODICE MIP 2797242

✓ **Alta definizione**
Immagini con risoluzione orizzontale superiore a 600 linee TV.

✓ **Elevato rapporto S/N**
Immagini sempre nitide grazie alla riduzione del disturbo ottenuta tramite speciali filtri digitali.

✓ **Fedele riproduzione dei colori**
Colori sempre brillanti e intensi con qualsiasi tonalità di luce.

cod. CAMCOLBUL27
€ 110,00

TELECAMERA IR a COLORI con DSP SONY EFFIO 600 LINEE TV

- Sensore da 1/3" CCD HR;
- Sensibilità 0,1 lux (F2.0);
- 753x582 pixel (PAL);
- Ottica f=3,6mm / F2.0;
- Angolo di ripresa 92,6°;
- 21 LED IR (portata ~15m);
- Alimentazione 12Vdc;
- Dimensioni 140x56x80mm;
- Da esterno (IP67);
- Completa di staffa.

cod. CAMCOLD19
€ 105,00

TELECAMERA DOME a COLORI con DSP SONY EFFIO 600 LINEE TV

- Sensore da 1/3" CCD HR;
- Sensibilità 0,05 lux;
- 753x582 pixel (PAL);
- Ottica f=3,8mm / F1.5;
- Angolo di ripresa 85°;
- Alimentazione 12Vdc;
- Dimensioni Ø125x85mm.

cod. FR475
€ 190,00

Prezzi IVA inclusa.

FUTURA ELETTRONICA
Via Adige, 11 • 21013 Gallarate (VA)
Tel. 0331/799775 • Fax. 0331/792287

Maggiori informazioni su questi prodotti sono disponibili sul sito *www.futurashop.it* tramite il quale è anche possibile effettuare acquisti on-line.

di MAURIZIO DEL CORSO

50 anni della CEP

La CEP, calcolatrice Elettronica Pisana compie 50 anni. Un bellissimo esempio di archeologia informatica senza la quale probabilmente oggi non esiterebbero i moderni TabletPC

Il 13 novembre 1961 si inaugurava a Pisa la CEP, Calcolatrice Elettronica Pisana. Non pensate ad una calcolatrice tascabile dei tempo moderni; la CEP era un macchinario alto due metri che occupava la superficie di un campo da tennis per un peso di 100 tonnellate. Il suo progetto nacque quasi per caso, a seguito di un finanziamento delle province di Pisa, Lucca e Livorno per la realizzazione di un Sincrotrone che poi fu costruito a Frascati. L'audace Enrico Fermi propose di utilizzare allora questi fondi per realizzare una macchina calcolatrice da utilizzare per scopi di ricerca e da lì nacque la CEP che poi fu la base dell'evoluzione delle macchine Olivetti. Erano quelli i tempi in cui i cervelli italiani non erano in fuga ma lavoravano a ritmi incessanti costruendo il futuro dell'informatica. Per ricordare questi eventi è stato allestito, a Pisa, un museo archeologico dell'informatica in cui è possibile rivivere in prima persona la realizzazione della CEP. Il museo sarà visitabile fino a giugno 2012 e magari noi di Fare Elettronica ci faremo un salto e vi mostreremo le "chicche" più interessanti. Intanto per chi volesse approfondire l'argomento, può seguire l'interessante documentario accessibile mediante il QRcode di questa pagina. Chissà quale faccia farebbe oggi Enrico Fermi nel sapere che tutta la potenza di calcolo della sua CEP può stare in un chip di 1 centimetro quadrato dal peso di qualche grammo…!

in fatto di "campagne"... la nostra non ha rivali

ABBONATI SUBITO

FINO AL 40% DI SCONTO !!!

Scopri gli altri abbonamenti in offerta qui

abbonamenti.ieshop.it

e la PROMOTUTTE3

fare elettronica

DIRETTORE RESPONSABILE
Antonio Cirella

DIRETTORE TECNICO
Maurizio Del Corso

Segreteria di redazione
Giorgia Generali

Comitato Scientifico
Simone Masoni (Microtest), Francesco Picchi (Microtest), Massimo Rovini (Università degli Studi di Pisa).

Art Director
Patrizia Villa

Hanno collaborato in questo numero:
Silvano Breggion, Daniele Cappa, Francesco Di Lorenzo, Giovanni Di Maria, Antonio Giannico, Massimiliano Miocchi, Giorgio Ober, Francesco Pentella, Mariano Severi.

Direzione Redazione
INWARE Edizioni srl
Via Giotto, 7 - 20032 Cormano (MI)
Tel. 02.66504755
Fax 02.66508225
info@inwaredizioni.it
www.inwaredizioni.it
Redazione: fe@inwaredizioni.it

Pubblicitá per l'Italia
Agostino Simone
Tel. +39 347 2230684
media@inwaredizioni.it

Europe and Americas
Elisabetta Rossi
Tel. +39 328 3245956
international@inwaredizioni.it

Asia
Cybermedia Communications Inc.
Tel. +886-(0)2-2691-2785
alex@cybermedia.com.tw

Rest of the world
Inware Edizioni srl
+39 02 66504755
info@inwaredizioni.it

Stampa
Tiber
Via della Volta, 179 - 25124 Brescia

Distribuzione
Parrini & C s.p.a.
Via di Santa Cornelia, 9
00060 Formello (RM)

Ufficio Abbonamenti
INWARE Edizioni srl
Via Giotto, 7 - 20032 Cormano (MI)
Per informazioni, sottoscrizione o rinnovo dell'abbonamento:
abbonamenti@inwaredizioni.it
Tel. 02.66504755
Fax. 02.66508225
L'ufficio abbonamenti è disponibile telefonicamente dal lunedì al venerdì dalle 14,30 alle 17,30.
Tel. 02.66504755
Fax 02.66508225

Abbonamento per l'Italia:
€ 55,00
Abbonamento per l'estero:
€ 115,00
Gli arretrati potranno essere richiesti, per iscritto, a **€ 9,00** oltre le spese di spedizione

Autorizzazione alla pubblicazione
Tribunale di Milano n.647
del 17/11/2003

Mensile associato all'USPI
(Unione Stampa Periodica Italiana)

© Copyright
Tutti i diritti di riproduzione o di traduzione degli articoli pubblicati sono riservati. Manoscritti, disegni e fotografie sono di proprietà di Inware Edizioni srl. È vietata la riproduzione anche parziale degli articoli salvo espressa autorizzazione scritta dell'editore. I contenuti pubblicitari sono riportati senza responsabilità, a puro titolo informativo.

Privacy
Nel caso la rivista sia pervenuta in abbonamento o in omaggio, si rende noto che i dati in nostro possesso sono impiegati nel pieno rispetto del D.Lgs. 196/2003. I dati trasmessi a mezzo cartoline o questionari presenti nella rivista, potranno venire utilizzati per indagini di mercato, proposte commerciali, o l'inoltro di altri prodotti editoriali a scopo di saggio. L'interessato potrà avvalersi dei diritti previsti dalla succitata legge. In conformità a quanto disposto dal Codice di deontologia relativo al Trattamento di dati personali art. 2, comma 2, si comunica che presso la nostra sede di Cormano Via Cadorna 27, esiste una banca dati di uso redazionale. Gli interessati potranno esercitare i diritti previsti dal D.Lgs. 196/2003 contattando il Responsabile del Trattamento Inware Edizioni Srl (info@inwaredizioni.it).

Collaborare con FARE ELETTRONICA
Le richieste di collaborazione vanno indirizzate all'attenzione di Maurizio Del Corso (m.delcorso@inwaredizioni.it) e accompagnate, se possibile, da una breve descrizione delle vostre competenze tecniche e/o editoriali, oltre che da un elenco degli argomenti e/o progetti che desiderate proporre.

LE PIU' RECENTI TECNOLOGIE ELETTRONICHE

DISPONIBILI IN 24 ORE

Panasonic
LINEAR TECHNOLOGY
ANALOG DEVICES
NXP
EPCOS
OMRON
AVX
MICROCHIP
Amphenol
molex
TEXAS INSTRUMENTS
CREE
VISHAY
TE connectivity

Farnell supporta gli innovatori che guardano al futuro. Grazie a centinaia di nuove tecnologie aggiunte online ogni giorno consegnate in 24 ore, sarai sempre fra i primi a disporre delle tecnologie più recenti dai principali brand dell'elettronica.

www.farnell.com/new

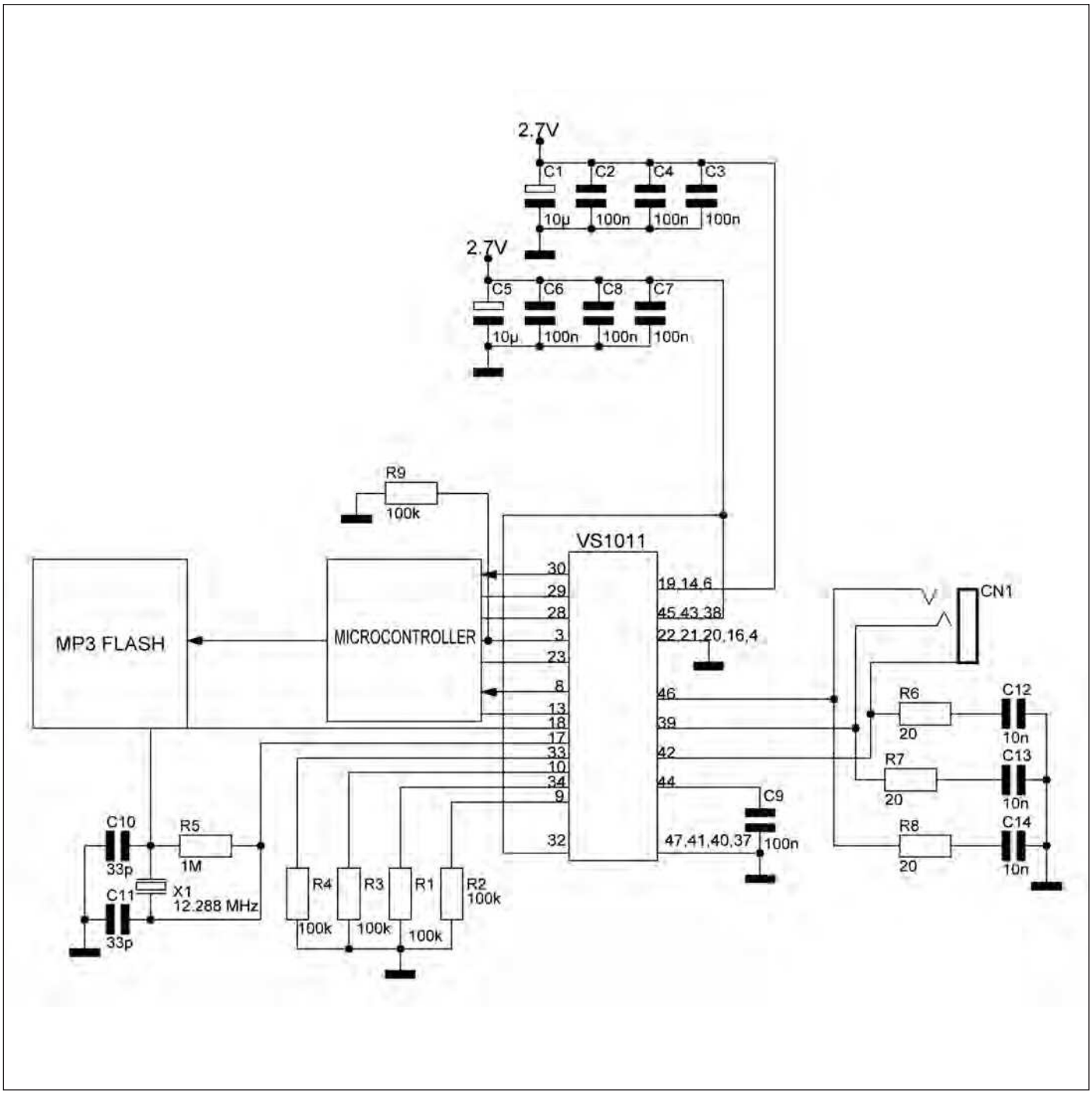
element14

www.element14.com

Farnell

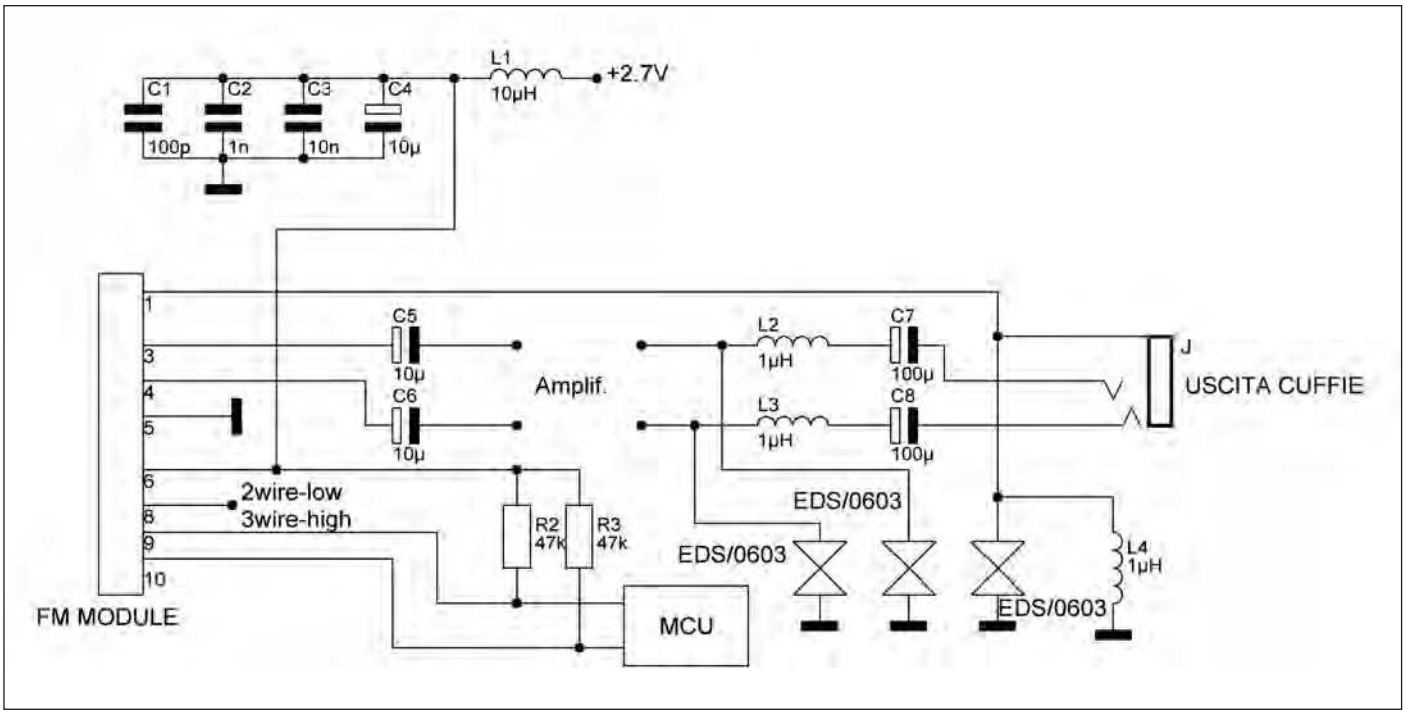
Progetta con i migliori

CODICE MIP 2807451



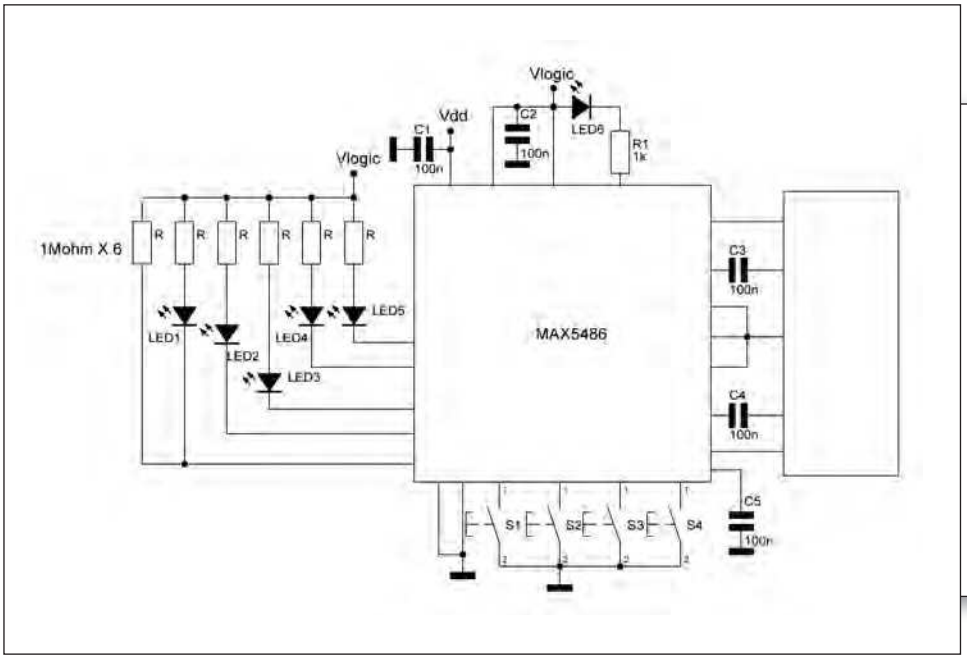
CODECMP3

Il VS1011, è un convertitore di file MP3. La gestione del file MP3 è garantita da una memoria RAM da 5K. L'integrato si presenta nella forma quadrata da 48 Pin. Viene alimentato con una tensione massima di 3.6V.



MODULO RICEVITORE FM

LM317 è un regolatore di tensione positivo che, collegato come in figura, fornisce una tensione variabile da 1.2V a 30V e una corrente costante di 5A. L'integratore realizzato con un lm741 garantisce il corretto valore di corrente, qualora il carico variasse. Il trimmer da 5K regola il flusso di corrente in uscita, mentre il trimmer da 200k regola la tensione di uscita.

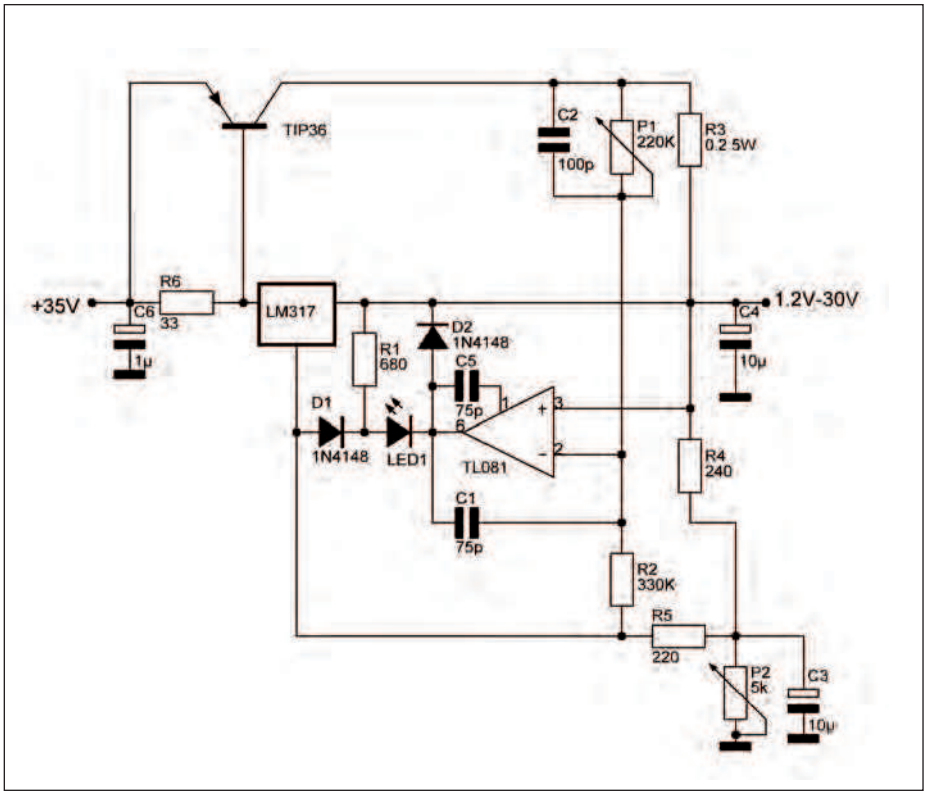


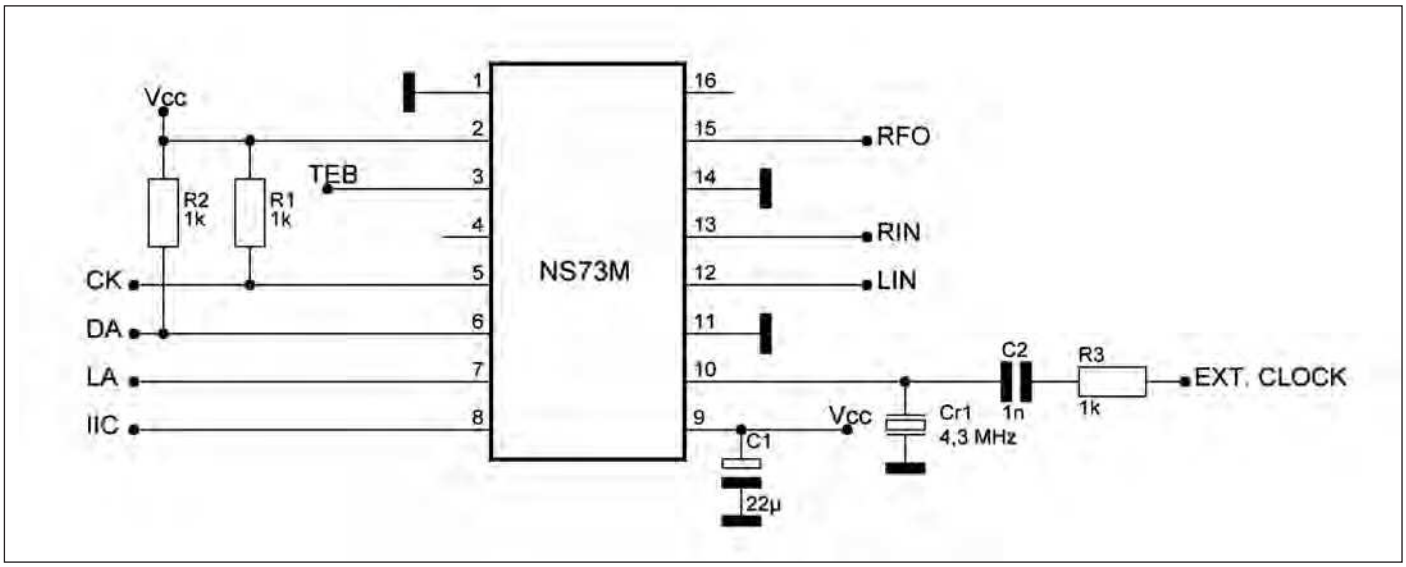
Controllo volume

Il MAX5486 viene prodotto dalla Maxim ed è impiegato per applicazioni di dispositivi portatili. Questo integrato permette il controllo del volume mediante un interfaccia costituita da pulsanti. Il tasto UP/BALL permette di incrementare il volume dell'uscita a sinistra mentre UP/BALR incrementa il volume del canale destro.

ALIMENTATORE VARIABILE

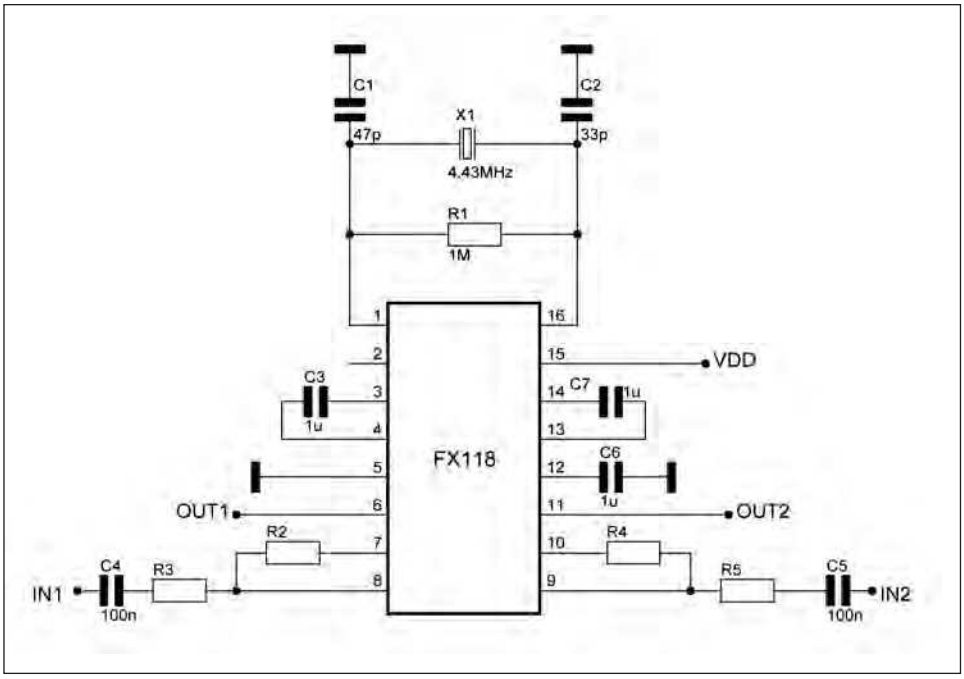
LM317 è un regolatore di tensione positivo che, collegato come in figura, fornisce una tensione variabile da 1.2V a 30V e una corrente costante di 5A. L'integratore realizzato con un lm741 garantisce il corretto valore di corrente, qualora il carico variasse. Il trimmer da 5K regola il flusso di corrente in uscita, mentre il trimmer da 200k regola la tensione di uscita.





TRASMETTITORE FM

L'NS73M è un modulo di trasmissione FM con larghezza di banda da 87.5 MHz a 108 MHz,viene impiegato in applicazioni come sistemi wireless, lettori MP3 ,ecc. Si presenta in un contenitore di alluminio per limitare la presenza di disturbi proveniente dall'esterno. Ha 15 pin, non richiede molti componenti esterni, e funziona con 3.6V.



SCRAMBLER RADIO

Il FX118 è uno scrambler ideato nel 1995 e aggiornato nell'ottobre del 2009. Lo scrambler è stato lo strumento adottato da molteplici apparati radio. Il compito di questo componente è quello di codificare le trasmissioni. Il consumo è ridottissimo e deve essere alimentato a 3V.

Massimizzate la vostra velocità di collaudo da 200 MHz fino a 1 GHz

CODICE MIP 2810366

Il nuovo WaveSurfer MXs-B è progettato per mettervi a disposizione le massime prestazioni nei casi in cui la velocità di elaborazione e l'accuratezza sono decisive. La memoria e la frequenza di campionamento migliori della sua classe assieme all'ampia gamma di trigger e matematiche avanzate, unite allo strumento di ricerca ed analisi WaveScan™, consentono agli ingegneri di trovare glitch ed eventi rari più rapidamente.

- Banda da 200 MHz a 1 GHz
- 10 GS/s –Migliore nella sua classe
- 25 Mpts di memoria standard – Migliore nella sua classe
- Fino a 36 canali digitali – Migliore nella sua classe
- Ricerca ed analisi WaveScan™
- La gamma di opzioni per bus seriali più ampia disponibile
- Generazione di report e documentazione LabNotebook™

Nuovi oscilloscopi WaveSurfer MXs-B

Disponibile presso:

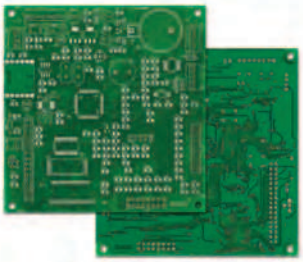
Vematron

Vematron S.r.l.
via Mons. Colombo, 34
21053 Castellanza (Va)
Tel. 0331 504064 Fax 0331 505380
www.vematron.it info@vematron.it

a partire da* 6970€

LeCroy
www.lecroy.com\it
Telefono 041-5997011

* WaveSurfer 42 MXs-B



I NOSTRI CIRCUITI STAMPATI HANNO DEI COMPONENTI UNICI. SU MISURA PER VOI.

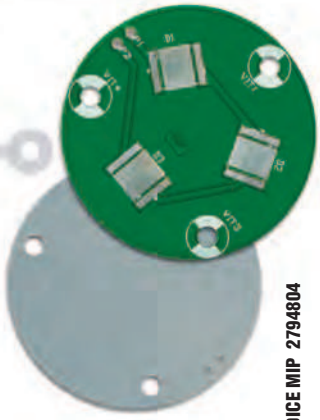
SINGOLA E DOPPIA FACCIA CON FORO METALLIZZATO; CON SOLDER E SERIGRAFIE PER UNO STAMPATO DI ALTA QUALITÀ O SOLO PISTE STAGNATE PER UN PROTOTIPO A BASSO COSTO

velocità
TEMPO DI CONSEGNA GARANTITO: 24 ORE O I CIRCUITI SONO GRATIS

qualità
I CIRCUITI VENGONO SOTTOPOSTI A TEST ELETTRICO E ACCOMPAGNATI DAL CERTIFICATO DI GARANZIA*; TUTTI I CIRCUITI SONO REALIZZATI SU LAMINATO Panasonic

novità
DA OGGI, ALLO STESSO PREZZO E ALLE STESSE CONDIZIONI DEI NOSTRI CIRCUITI DOPPIA FACCIA È DISPONIBILE IL LAMINATO IN ALLUMINIO AD ALTISSIMA DISSIPAZIONE TERMICA INDISPENSABILE PER LA REALIZZAZIONE DI PROGETTI CON LED AD ALTA LUMINOSITÀ E OVUNQUE SIA RICHIESTO UN ELEVATO GRADO DI DISSIPAZIONE

facilità
PER CHIARIMENTI, DETTAGLI SULLE NOTE TECNICHE, ORDINI:
www.mdsrl.it
PREVENTIVO ANONIMO, GRATUITO E IMMEDIATO



CODICE MIP 2794804

*la certificazione sarà rilasciata solo per circuiti con consegna oltre le 48 ore

md millennium dataware
produciamo circuiti stampati

Parco scientifico e tecnologico
15050 Rivalta Scrivia - Tortona (AL)
tel. 0131 860.254 fax 0131 860157 info@mdsrl.it

SCANALOGIC-2:
grandi caratteristiche ad un prezzo minimo!

Ikalogic ha annunciato il suo nuovo Logic Analyzer a 4 canali, facile da utilizzare e ricco di tutte le caratteristiche necessarie per facilitare gli sviluppatori di elettronica. Questo nuovo analizzatore logico consente, in maniera veloce e intuitiva, il campionamento dei protocolli seriali, la decodifica, il debugging e l'analisi dai più comuni bus di comunicazione (come UART, SPI, I2C, 1-wire, CAN, LIN ecc.). Nello Scanalogic-2 è inclusa la funzione "playback" dei dati precedenti oppure generare dei propri segnali in UART o PWM per testare i progetti o le apparecchiature. Il software di Scanalogic è stato progettato per essere facilmente utilizzato da studenti, ricercatori o semplici hobbisti. Inoltre, può anche generare i grafici in modo chiaro e semplice, per poi essere inclusi nei propri report.

CODICE MIP 2813630

Nuove telecamere quad-tap a 2 megapixel

Image S presenta due nuovi modelli di telecamere industriali con sensore quad-tap a 2 megapixel di JAI, in grado di acquisire immagini ad alta risoluzione a velocità (frame rate) superiori alle normali telecamere a CCD. I nuovi modelli AM-200CL (monocromatico) e AB-200CL (a colori) offrono una risoluzione da 1600x1200 pixel con frame rate standard da 17 fps. Le telecamere sfruttano l'architettura quad-tap nel sensore CCD KAI-02050 per l'acquisizione rapida di immagini, combinando poi le quattro sorgenti in due uscite per l'acquisizione ad alta velocità di immagini monocromatiche o a colori con matrice Bayer. Per semplificare l'integrazione, i due tap vengono elaborati tramite multiplexing mediante un data clock a 80 MHz per ottenere un'uscita a 8, 10 o 12 bit su una configurazione base Camera Link monocavo. Le telecamere offrono scansione dell'area di interesse (AOI) configurabile dall'utente, diverse modalità di binning (AM-200CL) e diverse modalità di acquisizione. Le funzionalità avanzate di pre-elaborazione delle immagini comprendono controllo automatico del guadagno e dell'otturatore dell'ottica, autobilanciamento dei bianchi, compensazione dei pixel difettosi, FFC e tabella di lookup a 256 valori.

CODICE MIP 2813637

Nuovo microcontroller da Renesas

Renesas Electronics, il primo fornitore di soluzioni di semiconduttori avanzati, ha annunciato il nuovo microcontroller a 16 bit altamente integrato, 78K0R/LG3-M, che consente la progettazione di rilevatori di cariche elettrostatiche monofase altamente integrati a costi contenuti. Questo nuovo microcontroller dalla alte performance associa un core a 16 bit a 20 megahertz (MHz) con 128 kilobyte (KB) di memoria Flash e 7 KB di RAM. Il 78K0R/LG3-M integra tutte le principali funzioni necessarie ad un rilevatore di cariche elettrostatiche, incluso un circuito dedicato al calcolo della potenza elettrica, un convertitore analogue-to-digital (A/D), un driver LCD, un real-time clock (RTC) con alimentatore indipendente. Questo nuovo prodotto di Renesas si rivolge principalmente ad applicazione di smart meter convenzionali, inoltre può essere utilizzato assieme ad altri prodotti della power-line Renesas e soluzioni di comunicazione wireless per creare, appunto, smart meter ottimizzati e dai bassi costi. Attraverso campagne di sensibilizzazione, i consumatori sono diventati più consapevoli del loro consumo di energia elettrica e l'importanza di ridurlo. La prossima generazione di smart meter consentirà la riduzione di energia elettrica grazie ad una comunicazione bidirezionale e display LCD integrati, che permetteranno la lettura a distanza, aggiornamenti tariffari e il bilanciamento del carico attraverso il controllo intelligente degli elettrodomestici.

CODICE MIP 2813641

Nuovo driver LED offline di Maxim

Maxim Integrated Products presenta il MAX16841, un driver per LED offline che garantisce la variazione luminosa senza sfarfallii, dalla massima intensità luminosa a intensità zero, sia con dimmer leading-edge (a triac), sia con dimmer trailing-edge (a transistor). Il controllo a frequenza costante ottimizza l'efficienza alle diverse tensioni di alimentazione della linea AC. Con un ampio range d'ingresso, da 90V a 264V AC, il MAX16841 costituisce una soluzione universale per l'impiego con i diversi modelli di dimmer. Consente la sostituzione diretta delle lampade a incandescenza e alogene, eliminando ogni problema di compatibilità nei confronti dei dimmer pre-installati. Il MAX16841 prolunga la durata delle lampade a LED, poiché è in grado di funzionare senza l'impiego di condensatori elettrolitici, comportando anche un risparmio di spazio e costi. E' ideale per il pilotaggio delle lampade regolabili a LED in retrofit alle tradizionali, e per il pilotaggio delle lampade a LED universali nelle applicazioni di illuminazione industriale, commerciale e residenziale.

CODICE MIP 2813587

e weekly
elektor newsletter

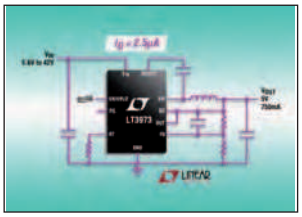
Iscriviti gratuitamente a E-weekly adesso

Vuoi essere aggiornato con le novità dal mondo dell'elettronica e dell'informatica?
Sei alla ricerca di suggerimenti, informazioni e offerte interessanti?
Iscriviti adesso a E-weekly, la newsletter gratuita di Elektor.

I tuoi benefici:
- le ultime news dell'elettronica nella tua casella di posta puntualmente ogni venerdì
- accesso gratuito all'archivio news sul sito Elektor
- sarai autorizzato a scrivere e ad aprire topics sul nostro forum.

Registrati oggi su
www.elektor.it/newsletter
anche dal tuo smartphone!

Regolatore switching step-down

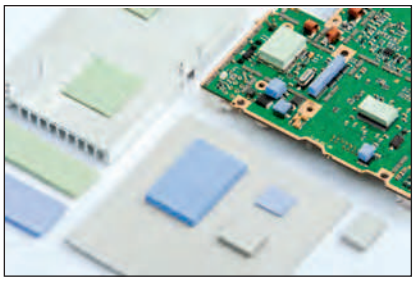


Linear Technology Corporation presenta l'LT3973, un regolatore switching step-down da 42V e 750mA con diodo di boost e catch diode integrati. Lo switch interno a 1,2A può fornire fino a 750mA di corrente continua in uscita con tensioni di soli 1,21V. Il funzionamento in Burst Mode mantiene la corrente di riposo al di sotto dei 2,5µA in condizioni di standby senza carico e sempre in questa modalità, il dispositivo fornisce bassissima corrente di riposo e pertanto è l'ideale per le applicazioni industriali e i sistemi automotive che richiedono circuiti sempre attivi e massima durata della batteria. Il range di tensioni in ingresso è tra 4,2V e 42V. L'LT3973 mantiene la tensione di dropout a livelli minimi (solo 530mV) quando la tensione in uscita scende al di sotto di quella programmata, ideale in condizioni come l'avviamento a freddo nelle applicazioni del settore automotive. L'utente può programmare la frequenza di commutazione tra 200kHz e 2,2MHz, consentendo così ai progettisti di ottimizzare l'efficienza ed evitare bande di frequenza sensibili al rumore.

CODICE MIP 2813643

Soluzioni concrete per la gestione della dissipazione del calore

Riuscire a dissipare al meglio il calore è un problema molto sentito da tutti coloro che producono e si trovano a gestire schede elettroniche. Come è facile intuire è necessario studiare un numero di soluzioni adeguato all'ampissimo numero di richieste possibili, lasciando sempre aperta la possibilità di customizzazione necessaria a rispondere positivamente ad ogni tipo di particolare necessità. Euro Technologies commercializza una famiglia di prodotti "Gap Filler": flessibili, elastici e in grado di coprire anche superfici irregolari, che vengono utilizzati ovunque sia richiesta un'alta conducibilità termica a bassa pressione. I Gap Filler di Euro Technologies, prodotti con uno spessore che va da 0,5 a più di 5 mm e conduttività termica da 0,8 a 6,0 W/mK, garantiscono ai progettisti la più alta flessibilità nella tolleranza dimensionale. L'estrema adattabilità riduce lo stress esercitato sui componenti mentre l'alta conduttività termica garantisce le performance termiche richieste.



CODICE MIP 2813645

addFuel.it

Radio amatore 2

13^ FIERA DEL RADIOAMATORE ELETTRONICA HOME COMPUTER

19-20 NOVEMBRE 2011

FIERA DI PORDENONE

CODICE MIP 2811845

fotomercato
Mostra-scambio internazionale dell'antiquariato, dell'usato e del digitale fotografico

2° MERCATINO DEL RADIOAMATORE

WWW.RADIOAMATORE2.IT

Pordenone Fiere
Fiera dell'Euroregione

FIERE ELETTRONICA
INFORMATICA E RADIANTISMO
con Mercatino Radio Amatoriale e Surplus
FIERA DI PIACENZA
PIACENZA EXPO' - LOCALITA' LE MOSE
17-18 DICEMBRE
ORARIO CONTINUATO 9.00 - 18.00
2012 PROSSIMI APPUNTAMENTI
CIRCUITO FIERE mondo elettronica.net
4-5 FEBBRAIO 2012
RAVENNA
PALAZZO MAURO DE ANDRE'
CODICE MIP 2798316
11-12 FEBBRAIO 2012
FIERA DI FERRARA
VIA DELLA FIERA 11 - FERRARA
DATA DA DEFINIRE
PIACENZA
PIACENZA EXPO' - LOCALITA' LE MOSE
DATA DA DEFINIRE
BOLOGNA
PALA NORD - PARCONORD
FARE ELETTRONICA
Tel. 0545 27548 - Cell. 348 9030832
www.mondoelettronica.net
Cosmohelp
VALE COME RIDOTTO ANCHE IN FOTOCOPIA

eventi

3-4 dicembre 2011

EXPOELETTRONICA FORLI'

Alla Fiera di Forlì appuntamento con uno dei più importanti eventi nazionali "consumer" di settore: quello con la Grande Fiera

Expo Elettronica

dell'Elettronica che da dicembre 2008 ha cambiato il nome in Expo Elettronica. Le cifre sono quelle delle grandi occasioni: circa 350 espositori, 20.000 mq di esposizione e circa 29.000 visitatori attesi alla manifestazione.

Dove: Forlì
Quando: 3-4 Dicembre 2011
Orari: 9.00-18.00
Organizzazione: Blunautilus
Info: www.expoelettronica.it

CODICE MIP 2799382

17-18 dicembre 2011

MARC di Genova

Al MARC saranno presenti numerosi espositori che rivenderanno prodotti di diverse categorie merceologiche: dalla componentistica elettronica a quella

MARC Mostramercato delle Attrezzature Radioamatoriali & Componentistica

informatica, dalla ricezione satellitare alla telefonia cellulare, periferiche da gioco e molto altro. Clicca sul pulsante sottostante o sulla voce di menù in alto per accedere all'elenco completo degli espositori e alle loro informazioni..

Dove: *Genova*
Quando: *17-18 Dicembre 2011*
Orari: *9.00-18.30*
Organizzazione: *Studio Fulcro*
Info: www.studio-fulcro.it

CODICE MIP 2813867

GRANDE FIERA DELL'ELETTRONICA

ERBA (CO)

presso Lariofiere

12-13 NOVEMBRE 2011

orario continuato dalle 09.00 alle18.00

Fiera dell'Elettronica in Genere

+ ASTRONOMIA

CODICE MIP 2800918

MANIFESTAZIONI FIERISTICHE
ITALFIERE S.R.L.

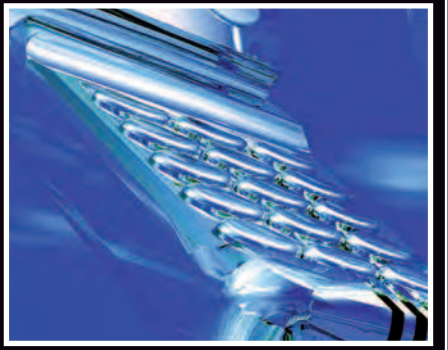
Puoi scaricare lo sconto sul biglietto d'ingresso:
www.italfiere.net

di FRANCESCO PENTELLA

Utilizzare ARDUIN con GSM/GPRS

Arduino rappresenta un classico esempio di architettura hardware open source, tanto da essere utilizzato in qualsiasi contesto operativo: dalle applicazioni con Bluetooth al Wi-Fi fino ad arrivare al GSM/GPRS

Esistono molteplici servizi di trasmissione dati offerti dal sistema GSM. In effetti il GSM, originariamente, ha voluto offrire servizi a commutazione di circuito con un data rate di 9600 bit per secondo per poi passare all'High Speed Circuit-Switched con 56 kibt/s.
Rispetto al sistema tradizionale, il nuovo HSCSD assegna diversi timeslot a ogni singola connessione introducendo, così, anche i nuovi servizi di bandwidth on demand e configurazione uplink-downlink asimmetrica. In questa configurazione di rete si preferisce parlare della commutazione di circuito, ossia la rete, una volta stabilita la connessione fisica da capo a capo fra i due utenti, dedica le proprie risorse fino a quando non ne viene esplicitamente richiesto il rilascio, indipendentemente dal fatto che i due utenti si scambino dati durante tutto il periodo della connessione. Un sistema di questo tipo è di certo performante se due utenti devono scambiarsi dati di una certa quantità ma, al contrario, può diventare inefficiente quando si decide di scambiare dati con minore entità o se lo scambio è di tipo bursty: in questo caso il tempo di utilizzo effettivo delle risorse di rete è solo una parte rispetto al tempo complessivo di connessione.
La tecnologia, nel tentativo di colmare questo gap, ha voluto suggerire una nuova modalità ricorrendo ad un servizio di trasmissione a commutazione di pacchetto o GPRS, General Packet Radio Service. In una classica connessione GSM, ovvero in un servizio a commutazione di circuito per la trasmissione dati, a ogni utente viene assegnato in modo esclusivo un canale di traffico, come sequenza di timeslot, che rimane a lui assegnato per tutta la durata della sessione. Le risorse sono così gestite secondo la modalità resource reservation, ossia sono impegnate fino al momento in cui non viene fatta esplicita richiesta di rilascio.
Nel GPRS si adotta la tecnica del context reservation, le risorse radio sono effettivamente impegnate solo quando c'è la necessità di inviare o ricevere dati. Le stesse risorse radio di una cella sono così condivise da tutte le stazioni mobili (MS) presenti in quella cella, aumentando notevolmente l'efficienza del sistema. In ambito GPRS i servizi possono essere caratterizzati dal fatto che la trasmissione di piccoli o grossi volumi di dati (ad esempio applicazioni interattive) è poco frequente o che la trasmissione di traffico dati bursty è intermittente di traffico dati bursty (ad esempio applicazioni in cui il tempo medio tra due transazioni consecutive è di gran lunga superiore alla durata media di una singola transazione).
Possiamo brevemente riassumere i diversi vantaggi della commutazione di pacchetto. In effetti, per prima cosa è così possibile mettere in multiplex più connessioni su un unico canale trasmissivo: in questo modo il canale diventa un mezzo condiviso da più connessioni. Non solo, in questo àmbito non si richiede l'instaurazione di un circuito dedicato tra sorgente e destinatario, quindi la fase di set-up della connessione risulta notevolmente velocizzata e, in ultima considerazione, è anche possibile eliminare le apparecchiature che, nel GSM base, fungevano da gateway per consentire l'interconnessione tra la rete GSM e le reti a commutazione di pacchetto.
La rete può riservare uno o più timeslot di una portante per il servizio GPRS: ogni timeslot, così configurato, costituisce un canale di traffico GPRS, chiamato anche Packet Data Traffic Channel. Su questo canale possono viaggiare, in trame diverse, pacchetti di utenti diversi; l'allo-



AT+SDATACONF	Configure the parameters of TCP/UDP connection
AT+SDATASTART	Start the TCP/UDP connection
AT+SDATATSEND	Send the TCP/UDP data(transparent, maximum length is 1460 bytes)
AT+SDATASEND	Send the TCP/UDP data(hex format)
AT+SSTRSEND	Send the TCP/UDP data(string format)
AT+SDATASTATUS	Query the socket status and the TCP sent/acknowledged/received bytes
AT+SDATAREAD	Read the received TCP/UDP data
AT+SDATATREAD	Read the TCP/UDP data(transparent)
AT+SDATARXMD	Set the received mode(hex or string format)
AT+TRT	Set the TCP maximum retry times

Figura 1: breve elenco di comandi AT.

Families & Modules	AT Interface Style supported (Factory Setting)		
GM Family (Modem)			
GM862-QUAD	0	1	2
GM862-QUAD-PY	0	1	2
GM862-GPS	0	1	2
GC Family (Compact)			
GC864-QUAD	0	1	2
GC684-QUAD /w SIM holder	0	1	2
GC864-PY	0	1	2
GE Family (Embedded)			
GE863-QUAD	0	1	2
GE863-GPS	0	1	2
GE863-PY	0	1	2
GE863- SIM	0	1	2
GE863- PRO^3	-	-	2
GE863- PRO^3 with Linux	-	-	2
GE864-QUAD	0	1	2
GE864-PY	0	1	2
GE864-AUTO	-	-	2
GT Family (Terminal)			
GT863-PY	0	1	2
GT864-QUAD	0	1	2
GT864-PY	0	1	2

Figura 2: estensione dei comandi.

cazione della banda ai singoli utenti è dinamica e dipende dal throughput istantaneo richiesto dall'utente e dalla banda disponibile. Il GPRS è utilizzato in qualsiasi rete GSM, cosa che permette a differenti applicazioni di essere veramente compatibile con Internet. Il GPRS assicura la possibilità di inviare pacchetti di dati verso qualsiasi destinazione con un data rate, di certo non interessante, compreso tra 9.6 Kbps e 150 Kbps. Tra le applicazioni più diffuse occorre citare i prodotti della serie Telit, tra cui il modello GM862-GPS. Tutti gli ultimi ritrovati della tecnica assicurano la possibilità di sfruttare le potenzialità GPS e GPRS attraverso comandi AT o anche SM5100B-D.

Insieme alle proposte di casa Telit esistono anche quelle di SEM che, al pari di Telit, assicurano ottime prestazioni con estrema flessibilità, considerato che è possibile sfruttare differenti modi operativi. I prodotti di ultima generazione permettono di utilizzare la tecnologia GPRS insieme allo stack TCP/IP. Nel caso delle proposte di SEM possiamo senza dubbio affermare che le varianti GM29, GM41 e GM47/GM48 non supportano lo stack, mentre, al contrario, le versioni GR47/GR48 e GT47/GT48 ne garantiscono il suo supporto.

In effetti, i comandi AT proposti dalla serie SM5100B-D offrono interessanti possibilità in ambito TCP/IP, tradizionalmente disponibili a livello socket con una stessa pila di un qualsiasi produttore commerciale, anche se, in questo àmbito, si offre maggiore flessibilità. I comandi via AT presenti, così come nei dispositivi Telit, consentono a qualsiasi applicazione di creare o eliminare socket UDP e TCP al fine di controllare la funzionalità GPRS e trasferire o ricevere dati verso qualsiasi modulo presente in rete. Non solo, i comandi AT assicurano anche la possibilità di interagire con la funzionalità TCP/IP arrivando anche a estrarre informazioni sul suo stato IP e sui diversi collegamenti attivi: la serie SM5100B-D offre la possibilità di gestire fino a dieci connessioni TCP. A questo scopo la **figura 1** mostra alcuni comandi AT utili in

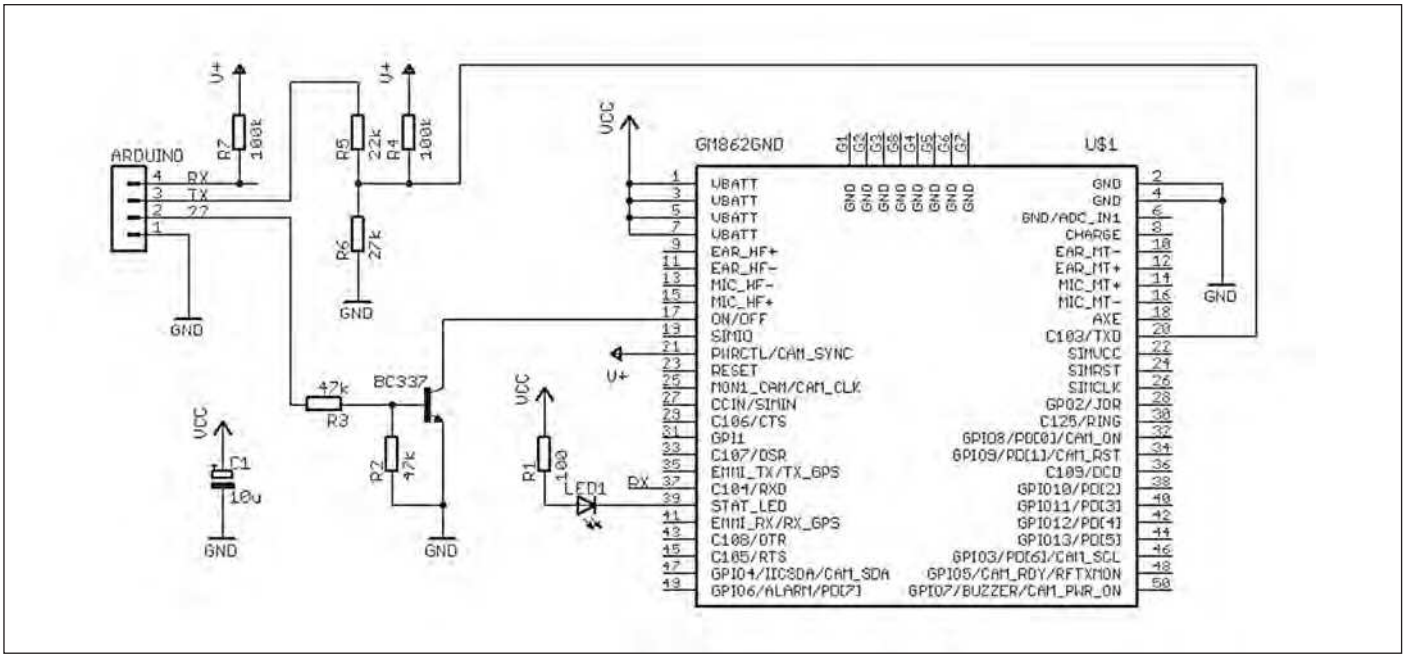


Figura 3: esempio di circuito tipico.

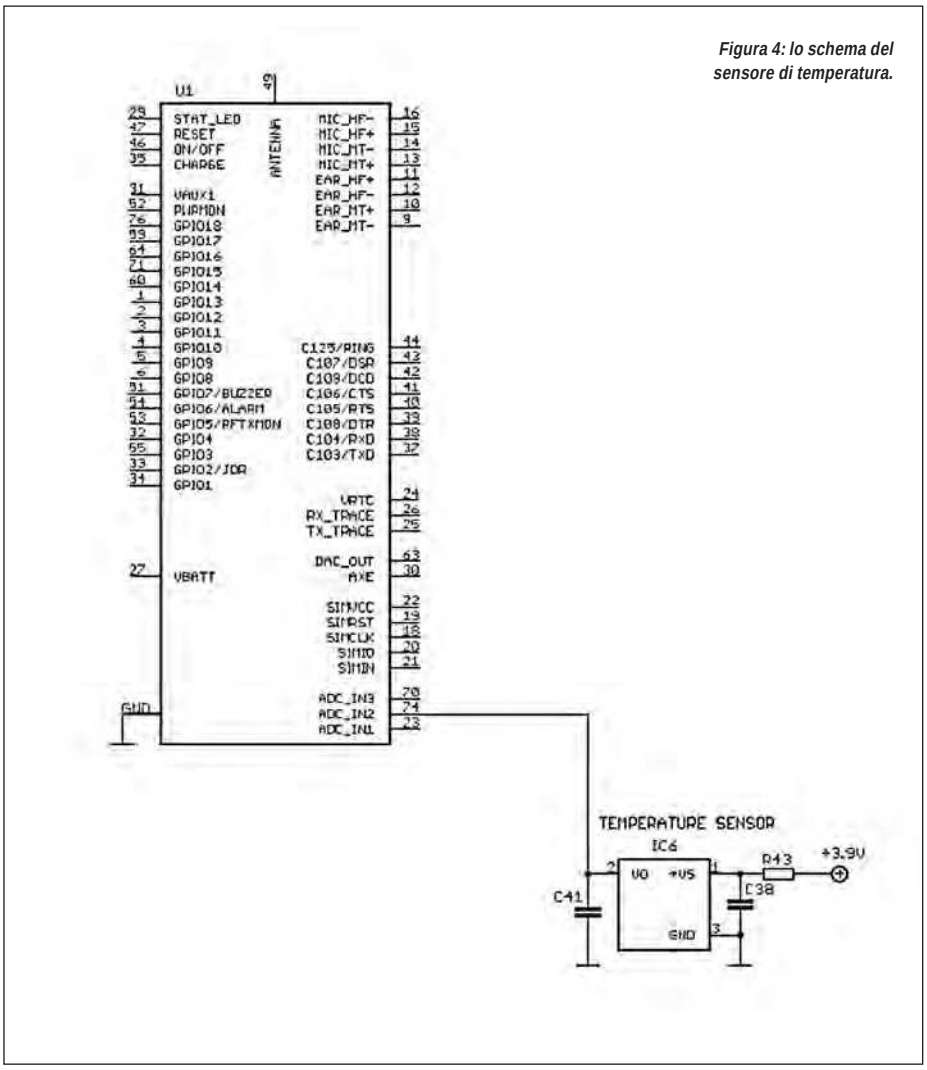


Figura 4: lo schema del sensore di temperatura.

questo contesto operativo e per maggiori dettagli sulle singole funzionalità può essere necessario riferirsi al manuale operativo. In particolare, il riquadro mostra l'accensione del modem della serie SM5100B-D con la relativa sessione iniziale. In ogni modo il sistema GPRS offre almeno due interessanti vantaggi: un migliore utilizzo delle risorse presenti e il supporto verso IP in modo completamente trasparente. La tecnologia utilizza risorse radio solo quando vi sono dati che devono essere inviati o ricevuti. La tecnologia a pacchetto consente alle applicazioni di occupare la rete solo quando un carico utile può essere trasferito e quindi ben si adatta alla natura stessa bursty di applicazioni dati. Un'altra importante caratteristica del GPRS è che fornisce una connettività immediata e un throughput elevato. La rete GPRS nasce proprio con lo scopo di assistere la rete GSM per tutto quello che riguarda il traffico dati a commutazione di pacchetto. La rete rappresenta un passaggio intermedio tra i precedenti sistemi di "telefonia" mobile (1G e 2G) e i nuovi sistemi di comunicazione personale (non solo telefonica) di terza generazione (3G) come l'UMTS. La rete GPRS permette di multiplexare più connessioni sullo stesso canale fisico: ogni risorsa è condivisa tra più utenti senza che venga assegnata in mo-

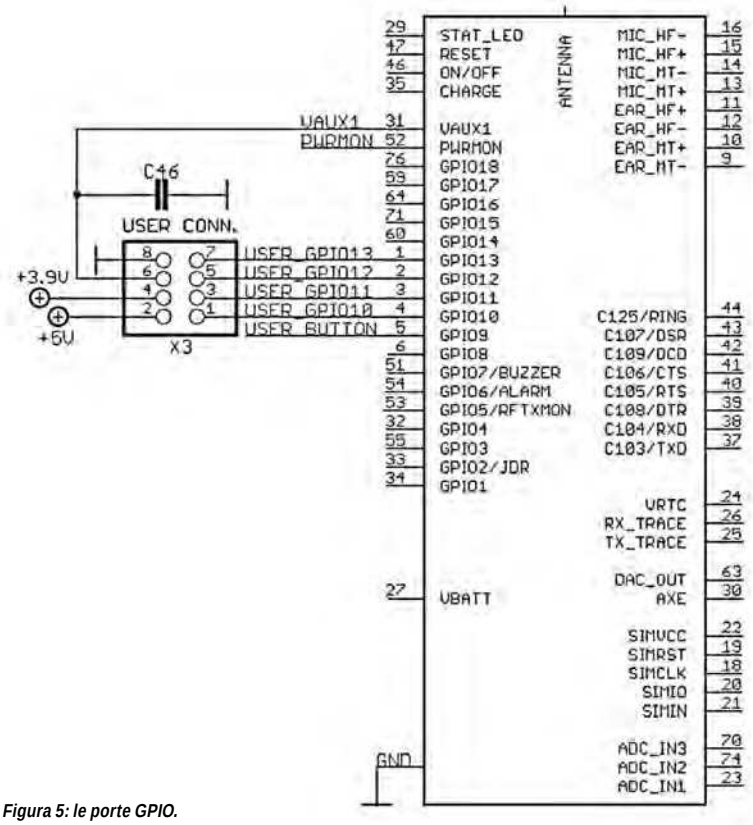


Figura 5: le porte GPIO.

do esclusivo, il che permette un uso molto efficiente della banda disponibile. Non solo, il GPRS prevede un'assegnazione flessibile delle risorse: a seconda delle necessità degli utenti è possibile allocare da 1 a 8 time slot sulla stessa portante in cui l'utente può trasmettere (il bit rate è compreso tra 9.05 e 171.2 kb/s).

IL SUPPORTO VERSO MODEM TELIT

Telit è uno dei protagonisti nel segmento dei dispositivi GSM/GPRS anche per via delle sue soluzioni che presentano un'alta integrazione di differenti funzionalità: dalla gestione di una seriale alla possibilità di pilotare attuatori fino alla possibilità di gestire direttamente il protocollo di rete. Tra i modelli più interessanti è possibile ricordare il Telit GM862, un completo modem GSM/GPRS quadriband per frequenze GSM a 850 Mhz, 900 Mhz, 1800 Mhz e 1900 Mhz (solo la nuova versione QUAD), ideale per le integrazioni di tipo OEM. Il dispositivo è compatibile con tutti gli altri moduli della stessa famiglia e l'aggiunta della caratteristica quad-band permette di poter integrare questo modulo in diverse applicazioni universalmente in tutto il mondo, approfittando delle sue caratteristiche quali il Jamming Detection, lo Stack TCP/IP o la Easy Scan Functionality. Attraverso il protocollo AT è possibile, attualmente, sfruttare tutte le sue funzionalità; in effetti, la comunicazione con il modulo GSM si basa su una

LISTATO 1

Inizializzazione modem

```
void GM862::init() {
  Serial.println("initializing modem ...");
  char buf[BUF_LENGTH];
  char cmdbuf[30] = "AT+CPIN=";
  strcat(cmdbuf, pin);
  requestModem("AT", 1000, true, buf);
  requestModem("AT+IPR=19200", 1000, true, buf);
  requestModem("AT+CMEE=2", 1000, true, buf);
  requestModem(cmdbuf, 1000, true, buf);
  state |= STATE_INITIALIZED;
  Serial.println("done");
}
```

TABELLA 1 – GPIO (GPIO10-GPIO13)

SIGNAL	FUNCTION	TYPE	I/O CURRENT	DEFAULT STATE	ON_OFF STATE	STATE DURING RESET	NOTE
GPIO10	I/O	CMOS 2,8V	1uA / 1mA	INPUT	LOW		
GPIO11	I/O	CMOS 2,8V	1uA / 1mA	INPUT	HIGH	HIGH	4,7k Pull Up
GPIO12	I/O	CMOS 2,8V	1uA / 1mA	INPUT	HIGH	HIGH	4,7k Pull Up
GPIO13	I/O	CMOS 2,8V	1uA / 1mA	INPUT	LOW		

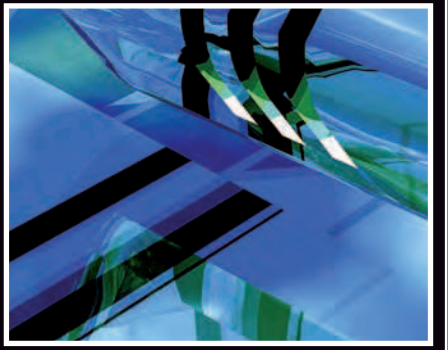
serie di comandi AT e, grazie a questo particolare protocollo, è possibile, per via di diversi parametri, indirizzare il componente e gestire la comunicazione con la rete. A questo proposito è possibile approfondire la materia leggendo la documentazione di riferimento del produttore, incluso il documento "GM862-QUAD / PY Hardware User Guide". In particolare, il modulo GSM della serie GE863 QUAD utilizza la versione V3 del chipset e questo garantisce una maggiore flessibilità e una estensione del suo set di riferimento **(figura 2)**. Per sfruttare tutte le prerogative dei dispositivi Telit di ultima generazione è necessario collegare il nostro Arduino alla porta seriale del componente allo scopo di inviare i diversi comandi via AT **(figura 3)**. Per fare questo è necessario collegare solo le linee TXD e RXD, mentre gli altri segnali di uscita del modem GM862 – quali DCD / pin 36, DTR / pin 43, DSR / pin 33, RTS / pin 45, CTS / 29 pin, e RI / pin 30 – possono essere aperti a meno che non si realizzi anche un controllo di flusso. Dalla figura si pone in evidenza che i pin logici del componente della Telit gestiscono solo valori di CMOS da 2,8 V: ecco perché è stato inserito un partitore di tensione sulla linea TX. Non solo, su entrambe le porte, Tx e Rx, sono state inserite resistenze di pull-up. Si è anche pensato di collegare un LED come indicatore di stato, scelta puramente opzionale, alla porta 39 del modem GM862: il pin 39 è un'uscita Open Collector a cui si può collegare direttamente un LED con una resistenza di pull-up, per visualizzare lo stato della rete e le informazioni sulla chiamata.

Il **listato 1** mostra una possibile inizializzazione del modem, mentre il **listato 3** pone in evidenza l'inizializzazione del lato GPRS con la sua abilitazione e, in seguito, grazie alla chiamata "AT#SK-

RIQUADRO 1- Sessione iniziale SM5100B-D

```
+Sind 1 Sim inserita
+Sind: 10,"SM",1,"FD",1,"LD",1,"MC",1,"RC",1,"ME",1 SIM
READY
+Sind 11 registriamoci al NETWORK
+Sind 3 Call Ready
+Sind 4 Sms Ready

-> Richiesta e aspettiamo l'accesso alla rete
AT+CGATT?
+CGATT : 1 (GPRS Attached)
OK
-> Setup PDP Context
AT+CGDCONT=1,"IP","IP a vostra scelta"
OK
-> Configura PDP context
AT+CGPCO=0,"user","password", 1
OK
-> Activate PDP Context
AT+CGACT=1,1
OK
-> Configura l'host remoto e la relative porta su con-
nessione TCP
AT+SDATACONF=1,"TCP","www.vostrascelta.com",80 (to a re-
mote host name)
OK
or
AT+SDATACONF=1,"TCP","211.123.11.21",1234 (to a remote
IP address)
OK
-> Start TCP/UDP connecting
AT+SDATASTART=1,1
OK
```



Pin	Signal	I/O	Function	Type	Input / output current	Default State	ON OFF state	State during Reset	Note
34	GPI1	I	GPI01 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		
28	GPO2	O	GPO02 Configurable GPIO	CMOS 2.8V	1uA / 1mA	OUTPUT	LOW		Open Collector Alternate function (JDR)
46	GPIO3	I/O	GPIO03 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	HIGH	HIGH	47K Pull Up
47	GPIO4	I/O	GPIO04 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	HIGH	HIGH	4.7K Pull Up Alternate function (RF Transmission Control)
48	GPIO5	I/O	GPIO05 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		Alternate function (RFTXMON)
49	GPIO6	I/O	GPIO06 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	fig. 01	HIGH	Alternate function (ALARM)
50	GPIO7	I/O	GPIO07 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		Alternate function (BUZZER)
32	GPIO8	I/O	GPIO08 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		
34	GPIO9	I/O	GPIO09 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		
38	GPIO10	I/O	GPIO10 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		
40	GPIO11	I/O	GPIO11 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	HIGH	HIGH	4.7K Pull Up
42	GPIO12	I/O	GPIO12 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	HIGH	HIGH	47K Pull Up
44	GPIO13	I/O	GPIO13 Configurable GPIO	CMOS 2.8V	1uA / 1mA	INPUT	LOW		

Figura 6: le porte GPIO su Telit.

LISTATO 2

```
requestModem

byte GM862::requestModem(const char *command, uint16_t
timeout, boolean check, char *buf) {

  byte count = 0;
  char *found = 0;

  *buf = 0;
  Serial.println(command);
  modem->print(command);
  modem->print('\r');
  count = getsTimeout(buf, timeout);
  if (count) {
    if (check) {
      found = strstr(buf, "\r\nOK\r\n");
      if (found) {
   Serial.println("->ok");
      }
      else {
   Serial.print("->not ok: ");
   Serial.println(buf);
      }
    }
    else {
      Serial.print("->buf: ");
      Serial.println(buf);
    }
  }
  else {
    Serial.println("->no respone");
  }
  return count;
}
```

TD=0,80,\"" risulta possibile stabilire un collegamento esterno per invio di messaggi oppure, con "AT$GPSACP", è anche possibile ricevere la posizione GPS. In realtà, con i moduli di Telit si possono inoltre acquisire segnali analogici o audio per incrementare le funzionalità a un costo, dalo punto di vista hardware, irrisorio. In effetti all'interno del componente è presente un convertitore A/D su 11 bit in grado di leggere tensioni da 0 a 2 V e convertirli in una rappresentazione su 11 bit, **tabella 2**.

Ad esempio, utilizzando l'LM61 **(figura 4)**, di National è possibile acquisire la temperatura esterna e gestirla sempre via AT. Il sensore sfrutta una scala lineare di 10 mV/°C nominale per un'escursione da –25°C a +85°C. La tensione proveniente dal sensore di temperatura è misurata dal modulo GSM sfruttando il comando che consente di leggere il valore analogico dal modulo, ovvero con AT#ADC=2,2,0 il modulo risponderà con il valore in millivolt. A questo punto la temperatura può essere ricavata in accordo a questa formula:

$$T[°C]=(VIN-600)/10$$

Ad esempio, così:

AT#ADC=2,2,0

Si ottiene

#ADC: 885

OK

LISTATO 3

```
Gestione GPRS

void GM862::initGPRS() {
  char buf[BUF_LENGTH];
  Serial.println("initializing GPRS ...");
          requestModem("AT+CGDCONT=1,\"IP\",\"inter-
net\",\"0.0.0.0\",0,0", 1000, false, buf);
  requestModem("AT#USERID=\"\"", 1000, false, buf);
  requestModem("AT#PASSW=\"\"", 1000, false, buf);
  Serial.println("done");
}
void GM862::enableGPRS() {
  char buf[BUF_LENGTH];
  Serial.println("switching GPRS on ...");
  requestModem("AT#GPRS=1", 1000, false, buf);
  Serial.println("done");
}
void GM862::disableGPRS() {
  char buf[BUF_LENGTH];
  Serial.println("switching GPRS off ...");
  requestModem("AT#GPRS=0", 1000, false, buf);
  Serial.println("done");
}
```

A questo punto la temperatura è uguale a

(885 – 600) / 10 = 28,5 °C.

Non solo, è anche possibile sfruttare le porte da GP10 a GP13 allo scopo di collegare un connettore esterno e utilizzare diversi dispositivi esterni, si veda la **figura 5**, mentre la **figura 6** pone in evidenza tutte le porte disponibili. In effetti, possiamo pensare di controllare quattro ingressi o uscite, in base alla sua configurazione, per gestire magari alcuni relé. L'applicazione esterna può essere alimentata direttamente dal connettore utilizzando le uscite da 5V/200mA compatibili con Arduino, da 3.9V/200mA con la stessa alimentazione del modulo GSM e, infine, da 2.85V/100mA direttamente dalla porta VAUX1 1 del componente di casa Telit.
A questo proposito la linea VAUX può essere controllata utilizzando lo stesso meccanismo del protocollo AT, così:

AT#VAUX=1,0 per disabilitare la linea VAUX1 in uscita;
AT#VAUX=1,1 utilizzato per disabilitare l'uscita;
AT#VAUX=1,2 è invece utilizzata per leggere lo stato della linea.

TABELLA 2 – A/D SU TELIT

	MIN	MAX	UNITS
Input Voltage range	0	2	volt
AD Conversion	-	11	bits
Resolution	-	<1	mV

Al contrario, le altre linee della GPIO possono essere gestite direttamente ricorrendo al comando:

AT#GPIO=n,n

Ad esempio, con il comando AT#GPIO=10,1,1 si imposta la porta di uscita 10 a un valore pari a 1, mentre con AT#GPIO=10,0,1 il pin viene azzerato. Per leggere il valore di un'uscita è necessario utilizzare il parametro 2, come

AT#GPIO=10,2

È anche possibile cambiare la direzione della porta con il comando AT#GPIO=10,0,0 da uscita a ingresso. A questo proposito la **tabella 1** mostra alcune caratteristiche di ciascuna porta.
Le possibilità sono diverse perché con l'ausilio del protocollo AT è possibile utilizzare diverse caratteristiche con un minimo impatto senza la necessità di scrivere moduli software. Non solo, con il comando AT#BND=0,1,2 o 3 è possibile selezionare la banda GSM che si intende utilizzare in accordo a questo schema:

0 - GSM 900 MHz + DCS 1800 MHz
1 - GSM 900 MHz + PCS 1900 MHz
2 - GMS 850 MHz + DCS 1800 MHz
3 - GMS 850 MHz + PCS 1900 MHz

Mentre con il comando AT+COPS? si ottiene il nome dell'operatore della telefonia mobile che si sta utilizzando.
Ricordiamo che per maggiori informazioni è opportuno consultare il manuale di riferimento di Telit, ovvero *AT Commands Reference Guide*. A questo proposito Old ich Horá ek ha messo a punto una libreria costituita da una serie di primitive che assicurano il controllo del componente, anche se, in realtà, è stata pensata per un'altra applicazione. La libreria scritta in C e facilmente portabile si basa sulla comunicazione seriale tra la scheda Arduino e il modulo sotto test attraverso l'uso di un buffer interno di 200 byte. La versione della libreria attualmente disponibile si basa su una comunicazione sincrona tra il processore e il componente, ovvero il programma rimane in attesa fino a quando il comando non si conclude. ❑

CODICE MIP 2805107

di DANIELE CAPPA

INSERITORE *magnetico*

Un inseritore per antifurto realizzato completamente con materiale di recupero che non utilizza un Pic! Per essere precisi non utilizza alcuna elettronica...

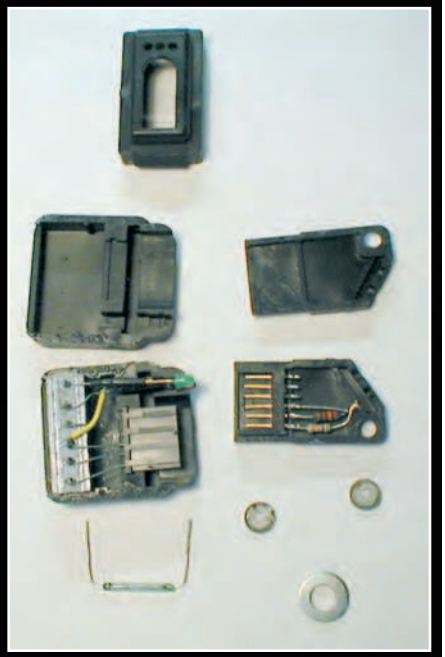


Figura 1: un vecchio inseritore smontato e il materiale necessario.

L'utilità di un progetto non è vincolata alla sua complessità, talvolta un oggetto semplice ha una versatilità che un suo omologo più performante non può raggiungere.
Una premessa è necessaria, si tratta di un progetto che presenta alcune particolarità che lo rendono molto originale, ma la cui sicurezza deriva interamente dal fatto che non è affatto ciò che sembra. Per cui è necessario, indispensabile, non divulgare la sua vera natura.
Lo scopo scatenante di questo progetto è un antifurto minimo, una protezione reale, non un semplice LED che simuli qualcosa che in realtà non esiste… Un impianto che deve proteggere un box, una cantina o un locale in cui sono riposti oggetti non particolarmente preziosi, ma che ugualmente gradiremmo non ci venissero sottratti. Con la consapevolezza che è inutile porre una porta blindata su un muro che può essere abbattuto in 10 minuti, ed è ugualmente insensato porre a guardia di due biciclette un impianto antifurto da alcune migliaia di euro.
Così com'è descritto, l'impianto è stato

Figura 2: la chiave modificata.

realizzato in un box della prima cintura torinese; è evidente che le foto proposte illustreranno solo i particolari, senza entrare nei dettagli. Il punto di partenza è rappresentato da un vecchio inseritore da antifurto, guasto ovviamente, da cui ricaveremo il guscio della chiave e la parte fissa, di solito si tratta di un frutto a dimensione standard per i più noti produt-

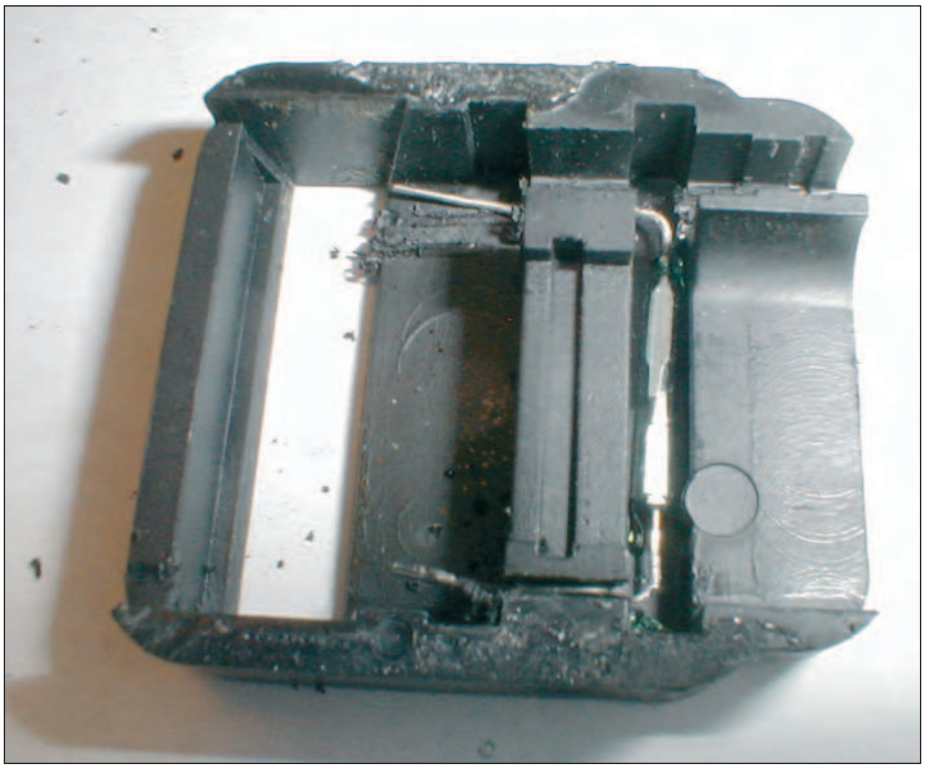


Figura 3: l'inseritore con il reed.

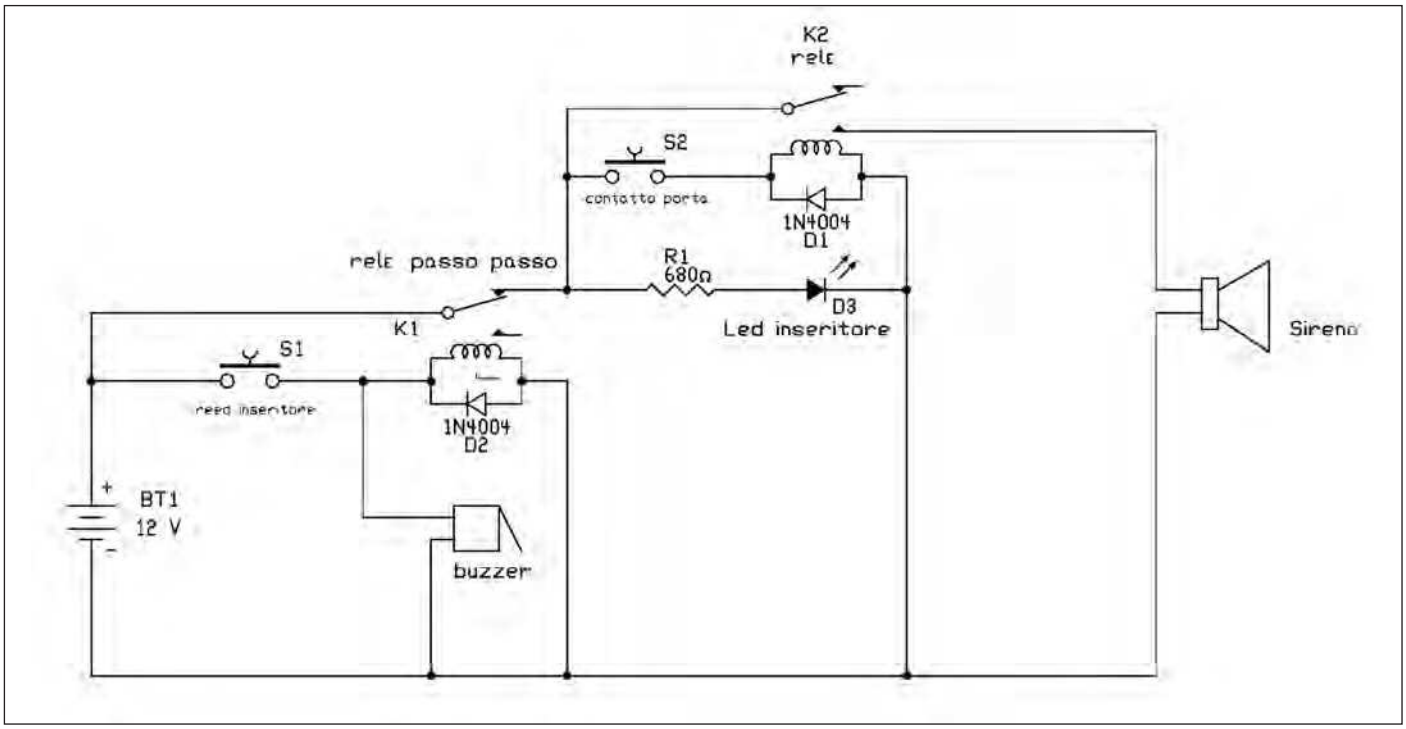


Figura 4: lo schema elettrico dell'impianto.

tori di serie di accessori elettrici (BiTicino, Gewiss ecc). Questa parte è provvista di LED che nell'impianto originale segnalavano le condizioni dell'impianto: se inserito, disinserito, in allarme ecc., e che possiamo tranquillamente riutilizzare.

LA MODIFICA

Il tipo di inseritore che più si presta alla modifica è quello in uso tra dieci e quindici anni fa; la chiave è formata da un piccolo connettore a cui fanno capo un paio di resistenze. All'interno della schedina di comando della chiave è presente un ponte che misura le resistenze appena inserite: se il valore è corretto la chiave "apre", in caso contrario viene emessa la segnalazione di chiave falsa e l'antifurto va in allarme. Per i nostri scopi è indispensabile la sola integrità fisica della piccola chiave. Elimineremo completamente quanto presente al suo interno, di solito due o tre resistenze, per sostituirlo con una piccola calamita. Anche questa è di recupero, ovviamente. In ogni lettore CD, o CD-rom, è presente una piccola calamita, si tratta apparentemente di una piccola rondella inserita al centro del centratore del disco (!). Questo particolare è solitamente abbastanza potente e sufficientemente piccolo, è insomma perfetto per i nostri scopi. Dopo aver localizzato la posizione in cui è possibile inserire la calamita nella ex chiave passiamo alla parte fissa, ovvero a inserire il sensore. Nel caso specifico si tratta di un'ampolla con un contatto reed che andremo a incollare in corrispondenza della posizione della calamita sulla parte mobile. Anche la parte fissa dell'inseritore andrà svuotata del suo contenuto, ma oltre a due o più LED di segnalazione, che possono comunque tornare utili, probabilmente non troveremo altro che la controparte del connettore della chiave.Quando inseriamo la chiave magnetica il contatto reed si chiude e il gioco è fatto, l'impianto dell'inseritore è completato da un relé passo passo del tipo a interruttore o a deviatore, secondo necessità, e preferibilmente alimentato a bassa tensione. Come conseguenza, a ogni inserzione della chiave il relé compie un "passo" e i contatti di uscita cambiano stato. Se lo riteniamo opportuno, in parallelo alla bobina del relé passo passo è possibile collegare un ronzatore a 12 V, un LED, oppure un più elegante cicalino che ci informi dell'avvenuta commutazione del relé. Nello schema elettrico è riportato sia il buzzer sia il LED che segnala che l'impianto è attivo (in merito sarà necessario considerare quanto esposto più avanti circa il consumo). L'elettronica del dispositivo è completata da alcuni diodi, posti in parallelo alle bobine dei relé, che si incaricano di sopprimere il picco di sovratensione che avviene all'atto dell'apertura del contatto che pilota le bobine. Questi diodi sono indispensabili, la loro assenza provoca una scintilla tra i contatti del reed che li degrada in brevissimo tempo. Vediamo ora le possibili caratteristiche del futuro impianto, alcune sono veramente particolari!

Così come è concepito, l'impianto vero e proprio impiega due relé, il passo passo comandato dall'inseritore attiva e disattiva il sistema mentre un relé normale comanda una sirena ed è a sua volta comandato da un semplice contatto magnetico posto sullo stipite della porta (o basculante) dell'ingresso del locale da proteggere. Il pregio di questo tipo di impianto è di avere, in fase di quiete, un consumo di corrente nullo. Se nel locale a cui è destinato l'impianto esiste una presa di corrente non ci sono problemi, un piccolo alimentatore a spina si occuperà di mantenere carica una delle "solite" batterie 12 V, 7 Ah, utilizzate normal-

Figura 5: il contatto montato sullo stipite della porta del box.

Figura 6: un altro inseritore già modificato.

mente per gli impianti antifurto che fornirà l'energia necessaria al funzionamento. In questo caso la condizione di antifurto inserito potrà essere evidenziata da uno dei LED presenti sulla parte fissa dell'inseritore e se il relé passo passo è del tipo a deviatore è possibile avere anche l'indicazione per l'impianto non inserito. Nel caso in cui il nostro locale *non* fosse fornito di energia elettrica le cose si farebbero un pochino più laboriose.
Non è possibile avere alcuna indicazione luminosa circa lo stato dell'impianto perché anche un solo LED scaricherebbe la batteria in circa due settimane e l'utilizzo di un LED lampeggiante potrebbe aumentarne un poco la durata. In queste condizioni è evidentemente necessario ricaricare la batteria periodicamente, ma l'impianto così come è concepito assorbe corrente dalla batteria solo al momento dell'inserimento, o del disinserimento, oltre che in caso di allarme ovviamente. Questo porta a una durata della batteria paragonabile alla sua autoscarica, e comunque una ricarica bimestrale potrebbe essere sufficiente. Il montaggio dell'apparecchio andrà eseguito a filo, non esiste alcun circuito stampato, solo i due relé e la batteria, che andranno collocati in una scatola adeguata, preferibilmente montata a muro e dotata di chiave.

REPERIRE IL MATERIALE

È stata utilizzata componentistica di provenienza automobilistica, ma tutto l'occorrente è facilmante reperibile anche in negozi di impiantistica civile e di elettronica. Una delle tante fiere radioamatoriali può fornire tutto il necessario mantenendo la spesa entro poche monete.
L'inseritore, la batteria e la sirena sono evidentemente di recupero, tutte provenienti da un vecchio impianto antifurto. La batteria nuova cosa meno di 20 euro e la sirena, oltre che nuova o in kit, è facilmente reperibile anche da installatori di impianti per auto, un esemplare usato potrebbe costare poco o nulla. Per l'inseritore è necessario rivolgersi a un installatore di impianti civili, sicuramente anche in questo caso non sarà difficile farsi regalare un oggetto che è comunque destinato al cassonetto dei rifiuti.
Se qualcuno avesse voglia di venirmi a trovare al lavoro sarò felice di barattare un ronzatore magnetico a 12 V con un caffè!
L'impianto esterno, ovvero i fili verso il reed presente nell'inseritore e verso il contatto magnetico della porta, rappresentano il punto debole del sistema. È necessario eseguire tutto con attenzione, utilizzando alcuni tubi per il passaggio dei fili e curando il cablaggio. Se il tutto risulta poco visibile avremo realizzato un sistema in grado di difendere il luogo da praticamente tutte i possibili malintenzionati. Appare evidente che una semplice calamita recuperata da un altoparlante è perfettamente in grado di agire sul reed di comando esattamente come la "chiave" modificata, per questo è essenziale che la vera natura dell'impianto non sia divulgata. Le probabilità che l'eventuale malintenzionato consideri questa possibilità sono praticamente pari a zero.
Impianti di questo tipo non hanno quasi bisogno di manutenzione, se si eccettua il controllo e la sostituzione della batteria che può tranquillamente avvenire ogni due o tre anni. I numerosi prototipi sono stati di volta in volta inseriti nei contenitori più diversi, da scatole provenienti da vecchi antifurto, ad armadietti di pronto soccorso con tanto di croce rossa, o a pensili recuperati dalla una vecchia cucina che praticamente tutti abbiamo nel garage. Se il lavoro esterno è ben curato e l'impianto risulta essere, a un primo esame, praticamente invisibile, allora abbiamo sicuramente dotato il nostro locale di un antifurto minimo che ci metterà al riparo dai ladruncoli, considerato che i professionisti non sono particolamnete interessati al contenuto delle nostre cantine o garage! Come sempre ringrazio chi si è prestato a "fare da cavia" e a Riccardo che mi ha fornito la materia prima, nello specifico alcuni vecchi inseritori smontati chissà dove. ❑

CODICE MIP 2805115

CAMPAGNA ABBONAMENTI

PROMO tutte 3

RISPARMIO ASSICURATO!
Risparmierai fino a 73,00 EUR sul prezzo di copertina. Garanzia del prezzo bloccato per un anno. Le riviste ti saranno recapitate comodamente a casa. Riceverai un buono sconto del 20% per un tuo prossimo acquisto su www.ieshop.it

UNA PROMOZIONE ESCLUSIVA:

- 11 numeri di FARE ELETTRONICA
- 11 numeri di ELEKTOR
- 11 numeri di FIRMWARE (edizione digitale) compreso l'accesso all'archivio a partire dal numero 49 di febbraio 2010
- 1 anno di accesso al CLUB di FARE ELETTRONICA
- 1 anno di accesso al CLUB di FIRMWARE,
- 20% di sconto per un acquisto su www.ieshop.it

TUTTO A SOLI € 119,50 ANZICHE' € 192,50

Sì, mi abbono alla PROMOTUTTE3 (è possibile abbonarsi online www.ieshop.it/promotutte3)

Riceverò 11 numeri di Fare Elettronica e 11 di Elektor in edizione cartacea, potrò accedere all'archivio di Firmware in edizione digitale a partire dal numero 49 di febbraio 2010 e ai prossimi 11 numeri, compreso un anno di accesso al CLUB di Fare elettronica e Firmware, tutto a **soli € 119,50** anziché €192,50

METODI DI PAGAMENTO

☐ **BONIFICO BANCARIO** ☐ **BOLLETTINO POSTALE** ☐ **ALLEGO ASSEGNO** (Intestato a Inware Edizioni)

☐ MasterCard ☐ VISA ☐ postepay

PayPal fare il pagamento su paypal@inwaredizioni.it

TITOLARE. . . . . .
NUMERO . . . . . .
DATA DI SCADENZA. . . . . CODICE DI SICUREZZA . . . . . .

INVIARE IL MODULO D'ORDINE A: Inware Edizioni srl, via Giotto 7 - 20032 Cormano (MI) - oppure via FAX allo 02 66508225
(Inviando questo modulo si accettano i termini e le condizioni riportate a tergo)

Nome . . . . . Cognome . . . . . .
Indirizzo . . . . . n° . . . . . Cap . . . . . .
. . . . . Città . . . . . Prov . . . . . .
Tel . . . . . Fax . . . . . Email . . . . . .
Ragione Sociale . . . . . Codice fiscale . . . . . Partita IVA . . . . . .

DATA . . . . . FIRMA . . . . . ☐ Fattura

1

PERCHÉ SCEGLIERE FARE ELETTRONICA:

LAVORO: Se lavori come progettista elettronico, Fare Elettronica saprà esserti utilissima con numerose idee e spunti sia dal punto di vista pratico che teorico. DIDATTICA: se lavori come insegnante in una scuola ad indirizzo tecnico, Fare Elettronica risulterà tra le migliori fonti di ispirazione per te e per i tuoi allievi. AGGIORNAMENTO TECNOLOGICO: se sei alla ricerca di una fonte d'ispirazione, Fare Elettronica non ti deluderà, fornendoti l'aggiornamento sulle più recenti tecnologie, trattate in modo esauriente e competente. HOBBY: se invece l'elettronica è la tua passione, Fare Elettronica ti permetterà di coltivarla giorno per giorno con progetti sempre appassionanti ed alla tua portata.

2

PERCHÉ SCEGLIERE FIRMWARE:

Firmware è una fantastica rivista digitale, tra le prime uscite nel settore dell'elettronica e la prima in assoluto in lingua italiana. Nasce come naturale evoluzione della rivista Firmware in edizione cartacea, rivista dedicata ai professionisti dell'elettronica, esigenti e competenti bisognosi di uno strumento di aggiornamento e di know how per l'elettronica professionale. Firmware è basata su una piattaforma che saprà regalarti una nuova esperienza di accesso a incredibili contenuti. Nessuna rivista ti sembrerà più all'altezza dopo aver visto Firmware!

3

PERCHÉ SCEGLIERE ELEKTOR:

Elektor è in circolazione in oltre 50 Paesi con un totale di 300,000 copie mensili. Pubblicato in Inghilterra, Francia, Paesi Bassi, Spagna, Grecia, Portogallo, Brasile, Svezia, Finlandia, Germania, Italia, Stati Uniti. La rivista Elektor è distribuita a livello mondiale tramite abbonamento, edicole e librerie. I siti internet sono tradotti in Inglese, Tedesco, Francese, Italiano, Cinese, Portoghese, Brasiliano, Svedese e Greco. 2 milioni di lettori mensili. 140.000 visitatori web mensili. 43.000 abbonati a e-weekly, la newsletter digitale. Elektor propone progetti di elevata qualità ideali per l'autocostruzione nei campi dei microcontrollori, audio high-end, RC modelling, RF e test & measurement. Elektor è creata da veri appassionati di elettronica. Elektor è l'unica rivista internazionale indipendente a vantare un laboratorio di progettazione e PCB design interno.

PERCHÉ SCEGLIERE TUTTE E TRE LE RIVISTE

Perché solo Inware Edizioni può darti una offerta così completa a condizioni così convenienti con un mix di aggiornamento, conoscenza, stimolo, divertimento e professionalità! Approfitta della fantastica PROMOTUTTE3 non te ne pentirai

di SILVANO BREGGION

Analizzatore *di rete* ENEL (V. G2)

Se non avete mai pensato di controllare il consumo di energia elettrica del frigo nuovo per capire se le caratteristiche di risparmio energetico tanto decantate dal venditore siano reali, oppure per conoscere il consumo della lavatrice per un determinato lavaggio o ancora quanto costa in termini di energia elettrica scaldare una pizza surgelata, ora potete farlo

Quello che vi presentiamo è uno strumento molto diffuso nell'ambiente industriale ma, nonostante il recente proliferare di dispositivi "made in China", poco conosciuto dalla maggioranza degli utenti casalinghi. È molto utile avere la possibilità di analizzare non solo l'impianto elettrico nella sua completezza, ma controllare con estrema semplicità anche un solo elettrodomestico. Gli esempi sopra citati sono solo alcune delle curiosità che si potrebbero soddisfare e, una volta che avrete in mano un analizzatore di rete ENEL, sicuramente troverete molte altre occasioni di impiego. Lo strumento in esame è in grado di misurare la tensione di rete, la corrente assorbita dall'impianto, il cos , cioè lo sfasamento tra la tensione e la corrente, la potenza istantanea e il consumo di energia elettrica, ovvero l'integrazione della potenza nel tempo.

Devo premettere che la precisione nella misura della corrente, e di conseguenza la potenza, non è accuratissima, soprattutto nei bassi assorbimenti, per via della definizione ampère/bit che presenta un rapporto piuttosto ampio, per non essere limitati dalla corrente massima che nell'impiego casalingo arriva a 17 ampère. La potenza minima misurabile è di circa 22 watt, quindi non pensate di controllare il consumo di una lampada a risparmio energetico perché l'unica informazione che riuscirete a leggere riguarderà la tensione. Per questo motivo la misura visualizzata nelle basse potenze non è molto affidabile, ma per potenze di 100-150 watt in su, i valori visualizzati migliorano e diventando sempre più precisi con l'aumentare dei carichi applicati. Per i più giovani può essere un ottimo esercizio didattico in previsione degli esa-

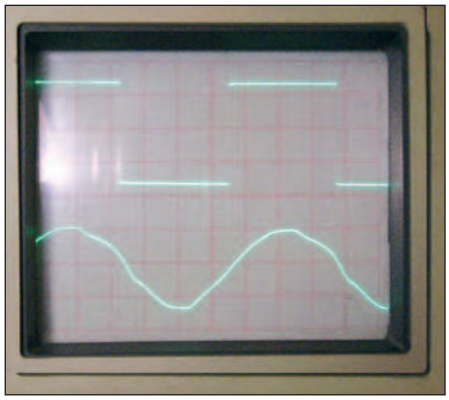


Figura 1. l'oscillogramma visualizza la sinusoide ai capi di R8 e l'onda quadra sul piedino 8 di U3c.

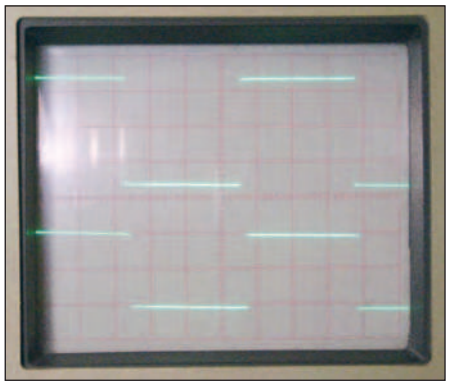


Figura 2: le due onde quadre sono in fase fra loro.

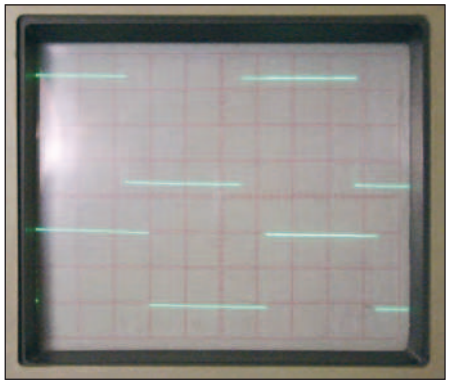


Figura 3: sfasamento tra la semionda della tensione e quella della corrente.

mi di Stato, per gli altri lettori l'inizio di un progetto in grado di elaborare funzioni e misure complesse che possono trovare impiego in applicazioni che vanno ben al di là dell'ambiente casalingo.

SCHEMA ELETTRICO

Lo schema elettrico è stato diviso in due parti, sia per semplificarne la spiegazione, sia perché la parte visualizzatrice (CPU) può trovare altre applicazioni cambiando solo il programma gestionale del PIC.
La prima parte del progetto riguarda l'alimentazione dell'intero dispositivo e lo stadio d'ingresso. Per calcolare la potenza è necessario misurare sia la tensione sia la corrente, quindi abbiamo bisogno di due ingressi analogici. Cominciamo dallo schema dalla misura della tensione che risulta il più semplice e intuitivo.
La presa d'ingresso, che può essere un cavo con una spina all'estremità, viene collegata ai morsetti 1 e 2 di X1. Poiché abbiamo a che fare con una tensione di 220 volt, dobbiamo ridurre la tensione a valori accettati dal PIC per essere misurati. A tal fine utilizziamo un semplice partitore resistivo composto da R1-R2-R3 e R4. Si è preferito usare tre resistenze (vedi R1-R2-R3) al posto di una soltanto unicamente per questioni di sicurezza. Infatti un'unica resistenza avrebbe avuto ai suoi capi una DDP molto alta, con la possibilità di causare archi all'interno della resistenza stessa soprattutto in presenza di disturbi o scariche elettrostatiche, sempre presenti nella rete elettrica. La presenza di tre resistenze limita la possibilità degli effetti sopra descritti, riducendo a valori accettabili la caduta di tensione ai capi di ciascuna di esse.
Quello che segue è un semplice raddrizzatore a singola semionda per applicare al PIC una tensione continua proporzionale alla tensione di rete. I condensatori C5 e C6, assieme alla resistenza R6, formano un'efficace filtro p greco in grado di livellare il segnale rendendo stabile la misura. V1 regola il corretto valore di tensione.
Dal partitore inoltre la tensione ridotta è prelevata da R7 e applicata all'ingresso non invertente di U3c.
Preferiamo descrivere questa parte del circuito nel capitolo dedicato alla misura del cosϕ. Una descrizione particolare merita la rilevazione e la misura della corrente. Per evitare insuccessi e delusioni, invece di costosi e introvabili trasformatori di corrente abbiamo preferito utilizzare la più semplice caduta di tensione su una resistenza incisa sul circuito stampato e assolutamente non critica. Due parole sui trasformatori di corrente. Certamente avrebbero reso più sicuro l'intero dispo-

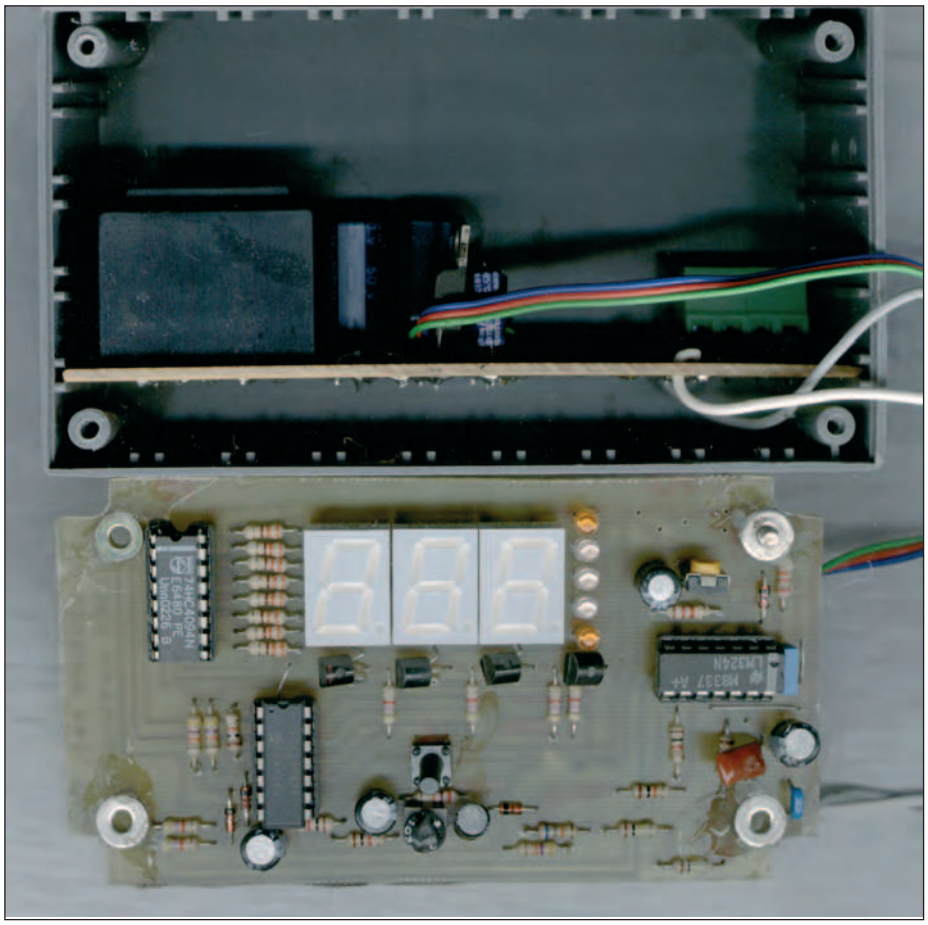


Figura 4: il contenitore e i circuiti.

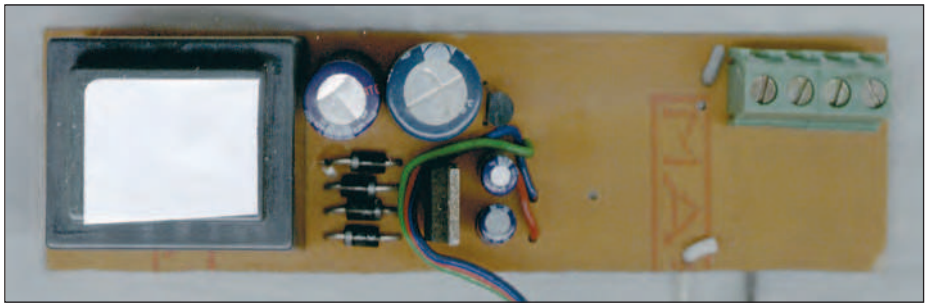


Figura 5: il circuito dell'alimentatore.

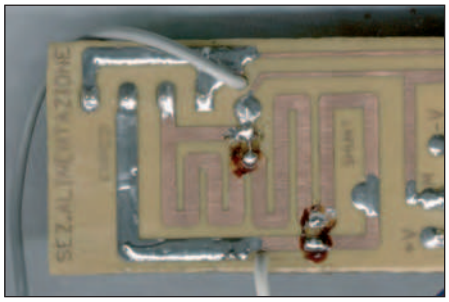


Figura 6: particolare della resistenza shunt sull'alimentatore.

Figura 7: il pannello frontale dell'analizzatore.

Figura 8: il display mostra la potenza istantanea in kilowatt.

Figura 9: il display mostra la tensione in volt.

Figura 10: il display mostra la corrente assorbita dal sistema in ampère.

Figura 11: il display mostra il valore di cos , cioè lo sfasamento tra la tensione e la corrente.

Figura 12: il display mostra il consumo di energia elettrica in kilowattora.

sitivo perché garantiscono l'isolamento galvanico, ovviamente aggiungendo un trasformatore all'ingresso dedicato alla misura della tensione.
Ma prove effettuate durante la progettazione di un precedente analizzatore di rete ci hanno consigliato di scartare tutti i toroidi che avevamo usato per la cattiva linearità. Solo un trasformatore di corrente recuperato da un inverter guasto ha dimostrato caratteristiche di precisione e linearità molto buone.
Certamente non possiamo chiedervi di smontare un inverter industriale per costruire un progetto a livello hobbistico, né proporre l'acquisto di un trasformatore appositamente progettato allo scopo, ben più costoso dell'intero progetto. Del resto con le soluzioni adottate in questo circuito, otteniamo il massimo della semplicità ed economia. Riprendiamo l'analisi del circuito d'ingresso dalla resistenza di caduta shunt.
Questa non è altro che una resistenza incisa direttamente sul circuito stampato e, dato il suo bassissimo valore, non provoca importanti cadute di tensione neppure in presenza di forti carichi. Nel caso abbiate la necessità di realizzare un dispositivo più compatto, è possibile autocostruire la resistenza di shunt semplicemente utilizzando uno spezzone di rame rigido del diametro di 0,8-0,9 mm, lungo circa 230 mm e avvolto a spirale su un diametro di 10-12 mm con i terminali piegati in modo da ottenere il montaggio verticale.
Ovviamente le spire devono essere spaziate tra loro in modo da non toccarsi e nello stesso tempo non tanto alte da superare l'ingombro del trasformatore di alimentazione in altezza.
Non dovete isolare il filo della resistenza in modo da permettere la dissipazione dell'energia sviluppata e tenere la temperatura della stessa la più bassa possibile.
Se optate per la resistenza autocostruita, dovete necessariamente aumentare il valore di R11 da 68 Kohm a 220 Kohm. La caduta di tensione ai capi della resistenza di shunt, che è proporzionale alla corrente che l'attraversa, risulta di pochi millivolt e va quindi amplificata. A questo pensa U3d nella configurazione invertente. Il guadagno è dato dal rapporto tra R11 e R10, circa 38 volte, mentre C7 ha il compito di attenuare tutte le frequenze superiori a 50 Hz.
Nota particolare per la configurazione circuitale di U3d: al fine di ottenere una precisione e una linearità maggiore all'aumentare della corrente, la soluzione migliore è quella dell'amplificatore differenziale, decisamente più complesso come circuito e per questo abbiamo preferito scartarlo per servirci del più semplice circuito amplificatore e restare fedeli alle linee guida che ci siamo prefissati durante la stesura del progetto.
Ai capi di R12 otteniamo la tensione amplificata che deve essere raddrizzata prima di venire applicata all'ingresso del PIC e misurata.
Il compito di tale funzione è affidato a U3a, configurato come raddrizzatore a singola semionda. Non possiamo usare un semplice diodo come per la misura della tensione perché le correnti deboli, anche se amplificate, non superano la tensione di soglia dei diodi al silicio che è di circa 0,6 volt.

LISTATO 1

```
_tx
    movlw     08h          ; Numero bit da trasmettere
    movwf      Count
_tx_1
    bcf       CLK          ; Clock (pin 3)
    btfss     buff,0       ; In “buff” il Byte da inviare alle uscite del 4094
    bcf       Data         ; Data (pin 2)
    btfsc     buff,0
    bsf       Data
    nop
    bsf       CLK

    rrf       buff         ; Rotazione a Dx per la lettura del bit successivo
    decfsz    Count,f      ; Deve essere ripetuto per 8 volte
    goto      _tx_1

            bsf            Strobe     ; Strobe (pin 1) trasferisce il Byte alle uscite del 4094
    nop
    bcf       Strobe       ; Condizione normale
    return                     ; Fine routine
```

Nel caso siano presenti due o più shift register, la routine di trasmissione dei dati può essere la seguente:

```
_tx
 ;          movlw  .8
;   movwf      Count
;   movf      buff_3,w        ; DATA per il terzo 4094
;   movwf     buff
;   call      _tx_1                  ; Trasmetti Byte 3
    movlw     .8
    movwf     Count
    movf      buff_2,w        ; DATA per il secondo 4094
    movwf     buff
    call      _tx_11                 ; Trasmetti Byte 2
    movlw     .8
    movwf     Count
    movf      buff_1,w        ; DATA per il primo 4094
    movwf     buff
    call      _tx_11                 ; Trasmetti Byte 1

    bsf       Strobe                 ; Trasferisce dati alle uscite dei 4094
    nop
    bcf       Strobe
    return                           ; Fine routine
_tx_1
    bcf       CLK                    ; Trasmissione di un BYTE per volta
    btfss     buff,7
    bcf       Data
    btfsc     buff,7
    bsf       Data
    nop
    bsf       CLK

    rlf       buff,f
    decfsz    Count,f
    goto      _tx_1
    return
```

La routine è stata testata usando come prova lo schema di figura 14. In “buff_1, buff_2, buff_3 ecc” verranno caricati i dati da inviare agli shift register. Il bit 0 di ogni Byte trasferito corrisponde al Q1 del 4094, mentre il bit 7 al Q8.

Siamo stati costretti a complicare un poco il circuito di misura. V2 servirà per regolare la corrente rilevata.
Anche in questo caso è stato posto in uscita un filtro a p greco costituito da C9 e C12 e dalla resistenza R15. Per alimentare l'intero circuito si è fatto uso di un piccolo trasformatore da 2 VA con due secondari da 8+8 Volt vista la necessità di alimentare l'operazionale con una tensione duale. Il regolatore U1 stabilizza la tensione positiva sia per la parte analogica sia per quella digitale, mentre U2 ha il compito di mettere a disposizione un tensione di –5 V rispetto alla massa per la sola sezione analogica. È sufficiente un regolatore plastico da 100 mA vista la richiesta modesta di corrente da parte di U3. La scheda principale viene alimentata per mezzo del connettore X3.

COME MISURARE IL COSΦ, CIOÈ LO SFASAMENTO TRA TENSIONE E CORRENTE

Durante le lezioni di elettrotecnica alle superiori, uno degli argomenti difficili da capire era proprio il cos . Si partiva da lontano, scomodando il cerchio trigonometrico e vettori in fase con la tensione, vettori in anticipo e, infine, vettori in ritardo, aumentando sempre più la confusione e, per quanto si studiasse, non scoccava quello spiraglio di luce che poteva illuminarci tanto da poter dire finalmente "ho capito". Immaginiamoci se eravamo in grado di misurarlo. Vediamo adesso come risolvere in maniera molto semplice il problema.
A questo punto abbiamo bisogno di trasformare in onda quadra la sinusoide distribuita dalle linee ENEL. Per la tensione ci pensa quella parte del circuito, di cui abbiamo accennato in precedenza, che si basa su U3c che non è altro che un comparatore di tensione. In sostanza tutto ciò che all'ingresso non invertente è maggiore del riferimento applicato all'ingresso invertente, in uscita risulta positivo.

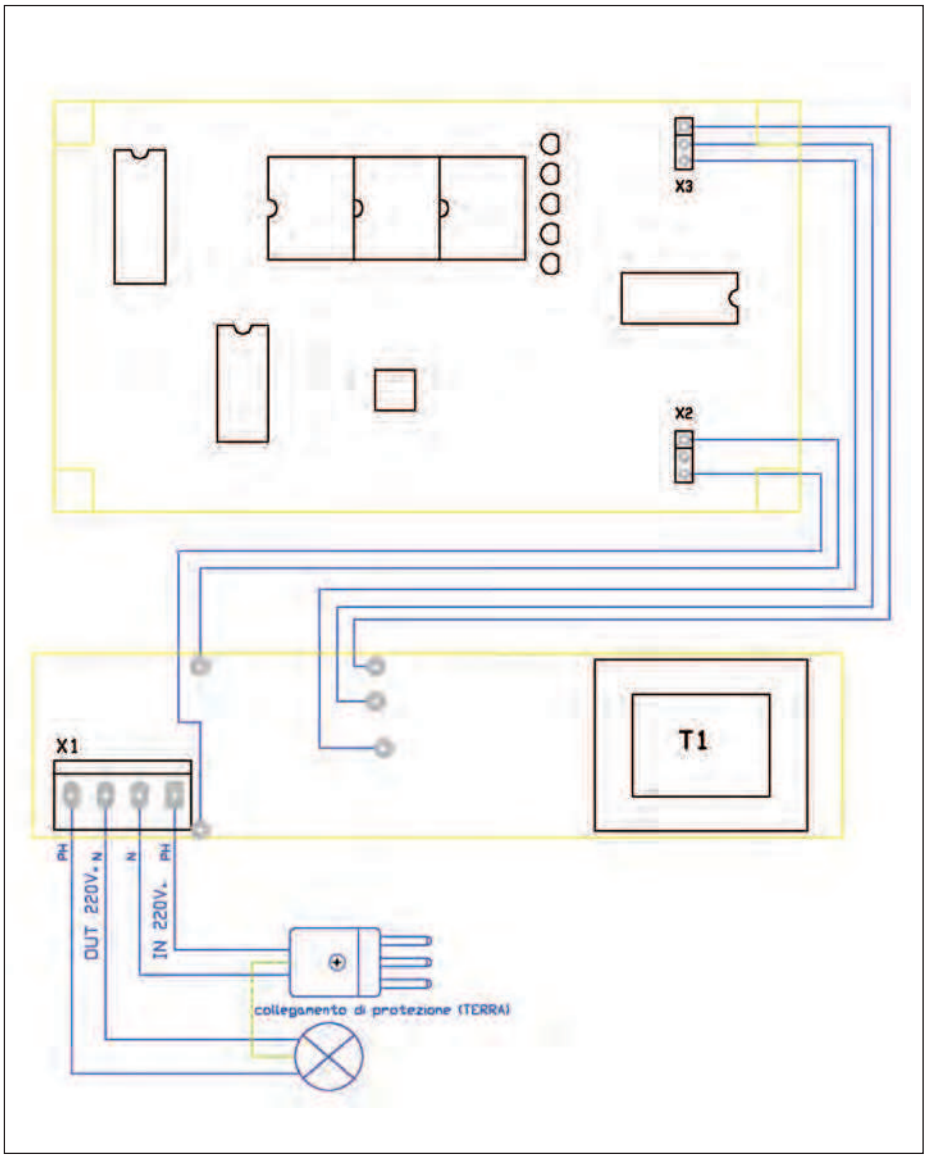


Figura 13: schema di collegamento.

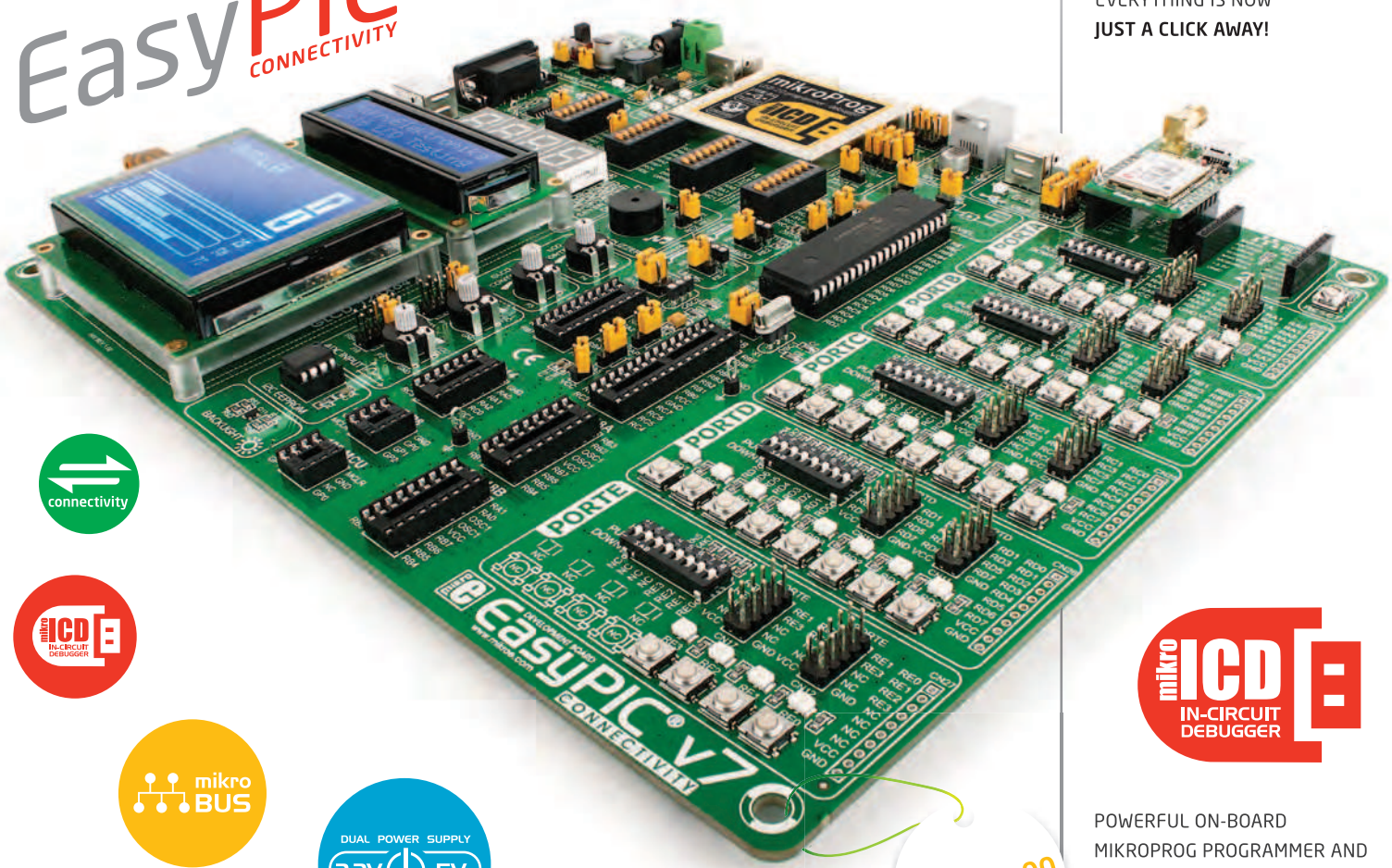
LISTA COMPONENTI

R1-2	100 KΩ 1/4 W
R3	68 KΩ 1/4 W
R4	47 KΩ 1/4 W
R5	27 KΩ 1/4 W
R6	1 KΩ 1/4 W
R7	10 KΩ 1/4 W
R8	1 KΩ 1/4 W
R9	4,7 KΩ 1/4 W
R10	1,8 KΩ 1/4 W
R11	68 KΩ 1/4 W (vedi testo)
R12	22 KΩ 1/4 W
R13	2,2 KΩ 1/4 W
R14	4,7 KΩ 1/4 W
R15	1 KΩ 1/4 W
V1	10 KΩ trimmer miniatura
V2	1 KΩ trimmer miniatura
C1*	470 µF 25 V elettrolitico
C2*	47 µF 16 V elettrolitico
C3*	470 µF 25 V elettrolitico
C4*	47 µF 16 V elettrolitico
C5	10 µF 16 V elettrolitico
C6	100 µF 16 V elettrolitico
C7	4,7 nF poliestere o ceramico
C8	680nF poliestere (vedi testo)
C9	100 µF 16 V elettrolitico
C10	100 µF 16 V elettrolitico
C11	0,1 µF poliestere
D1-4*	1N4004
D5-8	1N4148
U1*	7805 regolatore in TO220
U2*	79L05 regolatore plastico
U3	TL084
T1*	Trasformatore 230/8+8V 2VA
X1*	Morsettiera 4 poli per 220 V

I componenti contrassegnati dall'asterisco trovano posto nel CS della SHUNT.

IT'S HERE!

EasyPIC v7 CONNECTIVITY



connectivity

mikro ICD IN-CIRCUIT DEBUGGER

mikro BUS

DUAL POWER SUPPLY 3.3V 5V

$149.00

Best selling PIC development board in the world enters it's 7th generation of development. It is state of the art in design, functionality and quality. With 4 connectors for each port EasyPIC v7 has **amazing connectivity**. Ports are logically grouped with their corresponding LEDs and Buttons. Powerful on-board **mikroProg** In-Circuit Debugger and programmer supports over 250, both **3.3V and 5V devices**. Three types of displays, Serial EEPROM, two temperature sensors, Piezo Buzzer, USB connector, RS-232 and FTDI, Oscilloscope GND pins, as well as **mikroBus support** make this board an irreplaceable PIC development station.

mikro BUS

JUST PLUG IN ONE OF YOUR **CLICK BOARD**™, AND IT'S READY TO WORK. SPECIALIZED **MIKROBUS**™ **COMMUNICATION INTERFACE** MAKES YOUR DEVELOPMENT EASIER, AND ALLOWS SIMPLE, YET HIGHLY EFFECTIVE CONNECTIVITY. EVERYTHING IS NOW **JUST A CLICK AWAY!**

mikro ICD IN-CIRCUIT DEBUGGER

POWERFUL ON-BOARD MIKROPROG PROGRAMMER AND IN-CIRCUIT DEBUGGER CAN PROGRAM ALL **PIC10**, **PIC12**, **PIC16** AND **PIC18 MICROCONTROLLERS**. OUTSTANDING PERFORMANCE AND EASY OPERATION WILL BLOW YOUR MIND. YOU WILL NEED IT, WHETHER YOU ARE A PROFESSIONAL OR A BEGINNER.

DUAL POWER SUPPLY 3.3V 5V

EASYPIC V7 IS THE ONLY DEVELOPMENT BOARD IN THE WORLD TO SUPPORT BOTH 3.3V AND 5V MICROCONTROLLERS. REVOLUTIONARY ENGINEERING ALLOWED US TO SUPPORT **OVER 250 MICROCONTROLLERS** IN A SINGLE BOARD. IT'S LIKE HAVING TWO BOARDS INSTEAD OF ONE!

MikroElektronika
DEVELOPMENT TOOLS | COMPILERS | BOOKS

GET IT NOW
www.mikroe.com

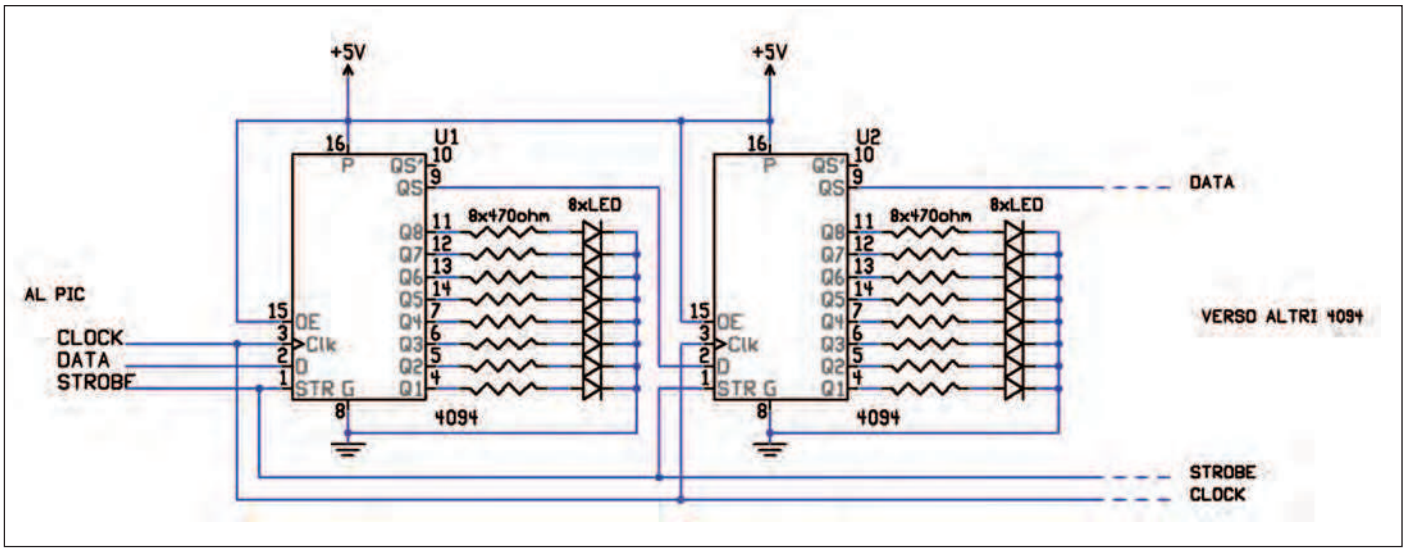


Figura 14: collegamento test.

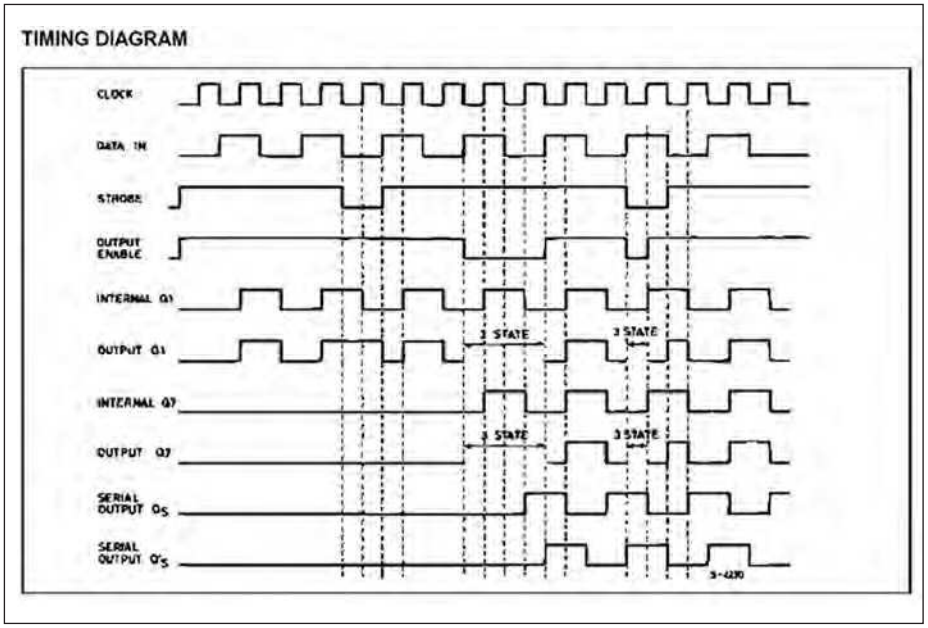


Figura 15: protocollo di accesso.

In pratica U3c è configurato come rilevatore di passaggio per lo zero e lo zero rappresenta il confine tra la semionda negativa e quella positiva e, ovviamente, viceversa. L'oscillogramma di **figura 1** visualizza la sinusoide ai capi di R8 e l'onda quadra sul piedino 8 di U3c. La stessa conversione di forma d'onda avviene anche su U3b, comparatore dedicato alla misura della fase della corrente (sinusoide ai capi di R12 e onda quadra sul piedino 7 di U3b).

Confrontando le due onde quadre **(figura 2)**, possiamo notare come queste siano in fase fra loro. In realtà la corrente è invertita di segno da U3d: ai fini della misura non cambia nulla, è sufficiente tenerne conto durante la programmazione del PIC.

Nella **figura 3** notiamo chiaramente lo sfasamento tra la semionda della tensione e quella della corrente, assorbita da un carico induttivo (o misto, lo sfasamento di un carico induttivo puro risulta di 90 gradi), in questo caso rappresentato dal motore della molla del laboratorio.

Bene, ma ora come si misura? Nella frequenza di rete, 50 Hz appunto, il periodo è di 1/50 di secondo, in altre parole 20 ms (20 ms = 0.02 s). Nel cerchio trigonometrico il periodo è un giro completo che vale 2 radianti, ovvero 360 gradi. Una singola semionda vale radianti, cioè 180 gradi, quindi 10 ms. Se dividiamo i 10 ms/180 gradi otteniamo 55 s (55 milionesimi di secondo). Imponendo al micro la partenza di un timer in presenza del fronte in salita della semionda positiva della tensione e l'arresto quando viene rilevata quella negativa della corrente, otteniamo il tempo dello sfasamento della corrente con la tensione che se non esiste vale zero, altrimenti un tempo che diviso per 55 s dà come risultato un valore espresso in gradi, cioè il ritardo della corrente con la tensione. Semplice no?

Dunque i gradi dello sfasamento sono misurati con un tempo, ma la formula del calcolo della potenza chiede il cos e non i gradi.

Senza scomodare complicate formule, è sufficiente ritrovare in una tabella il corrispondente valore in cosϕ dei gradi misurati (vedi la subroutine "_Calcolo_cosfi"). Ovviamente se i gradi sono zero, dalla tabella il cosϕ assume il valore 100 (usiamo 100 invece di 1,00 per comodità di calcolo), cioè cosϕ=1, nel caso di valori di gradi diversi, il cosϕ assumerà il corrispondente valore.

Nota importante al fine di ottenere la maggiore precisione possibile durante

la lettura del cos riguarda il condensatore C8. Il valore consigliato è il risultato di prove empiriche, nel montaggio potrebbe essere necessario un'aggiustamento del valore. In particolare, aumentando C8 lo sfasamento aumenta, cioè diminuisce il cos , e viceversa.
Se avete esagerato aumentando a dismisura la capacità, vi trovate con un cosϕ inferiore a 1,00 nonostante il carico sia di tipo resistivo. In tal caso è necessario ridurre il valore di C8. Altra considerazione pratica, il valore del cos diventa attendibile con l'aumento del carico. Già con 300 milliampère il valore misurato si avvicina a quello reale.

CALCOLO DELLA POTENZA E DEL CONSUMO DI ENERGIA

Potrà sembrare banale, ma la soluzione più semplice consiste nell'applicare brutalmente la nota formula: P=V*I*cosϕ. In questo modo si potrà visualizzare la potenza istantanea campionata ogni mezzo secondo circa.
La stessa potenza viene sommata a un registro a 16 bit a ogni campionamento per un minuto, e infine divisa per il numero di campionamenti effettuati nel minuto. In pratica sarà considerata la media dell'energia prelevata ogni minuto. Un minuto corrisponde a 1/60 di ora, la media dell'ultimo minuto deve essere divisa per 60 e il risultato va sommato al contatore dell'energia. Per rendere più precisa la misura, il resto dell'ultima divisione non va scartato ma aggiunto alla media del successivo minuto campionato.

SCHEMA CPU E DISPLAY

Come già detto, abbiamo preferito tenere diviso lo stadio d'ingresso da questa parte di schema perché riteniamo il sistema di pilotaggio dei display molto interessante ed economico da potere trovare applicazioni in altri progetti, ovviamente con le opportune modifiche del software.
Una delle difficoltà che incontriamo nella progettazione dei dispositivi gestiti da un microcontrollore è il numero di piedini di I/O, che non bastano mai. È vero che esistono modelli di PIC con 28, 40, o più piedini, ma costano parecchio e ci eravamo prefissati l'obiettivo di proporre uno strumento valido ed economico allo stesso tempo. La scelta è caduta sul PIC16F676, un microcontrollore di casa Microchip che costa meno di 3 euro, IVA compresa, e ha a disposizione una serie di periferiche da non far rimpiangere la mancanza di micro più costosi, compreso un ADC a 10 bit che fa proprio al caso nostro per la misura della tensione e della corrente. Esso dispone, inoltre, di 128 Byte di EEPROM da utilizzare nel salvataggio dell'energia prelevata.
Unico difetto: possiede "solo" 14 piedini. Per fortuna incorpora un oscillatore di tipo RC calibrabile e, tolti i due piedini per l'alimentazione, ne restano 12 da usare come I/O. Una piccola annotazione: questa famiglia di PIC esce dalla fabbrica con il valore della calibrazione dell'oscillatore interno memorizzato alla locazione 3FFh, ultima word di memoria. Al fine di evitare di perdere il contenuto durante la cancellazione accidentale, vi consiglio di leggere tale valore dal PIC e di annotarlo in un'etichetta da applicare al chip stesso prima della programmazione per ottenere la massima precisione dall'oscillatore. Ritorniamo al nostro schema. In un primo momento avevamo pensato di utilizzare un display LCD intelligente, ormai collaudato e ampiamente usato nei nostri progetti ma, oltre a essere costoso, la parte superiore è di metallo e collegata a massa. Con la 220 V presente nel circuito questo particolare ci ha scoraggiato dall'utilizzo perché aumentava di molto la possibilità di scosse elettriche. È stata quasi una scelta forzata quella del display a sette segmenti a LED a catodo comune, considerata la natura isolante del contenitore, inoltre è molto facile trovarne nel surplus o comunque nei mercatini.
L'unico svantaggio: la necessità di avere a disposizione diversi piedini di I/O per pilotare correttamente i tre display. Infatti ognuno di questi contiene 8 LED che, moltiplicati per tre display, richiedono 24 piedini che non abbiamo a disposizione. Non dobbiamo dimenticarci che dovremo riservare 4 piedini per la misura della corrente e della tensione e per misurare la fase tra le due e un piedino come ingresso per controllare la pressione del pulsante. Restano 7 piedini per pilotare 24 LED! Non resta che appoggiarsi a un integrato esterno e la scelta è caduta su un shift register di tipo CMOS, il 4094 o l'equivalente TTL 74HCT4094.

IL 4094 (74HCT4094)

Si tratta di uno shift register con caricamento seriale e uscite in parallelo definito con l'acronimo SIPO (Serial In Parallel Output). Vediamone il funzionamento perché si presta molto facilmente alla moltiplicazione delle uscite. È possibile anche utilizzare dei PISO (Parallel Input Serial Output) con lo scopo di moltiplicare gli ingressi del micro **(figura 14)**.
Dunque abbiamo a disposizione un ingresso e 8 uscite. Il Byte da trasferire nelle uscite viene inviato serialmente nei flip-flop del 4094 attraverso l'ingresso (pin 2) e sincronizzato dal clock (pin 3). Una volta caricato l'intero Byte, un impulso sul pin 1 (strobe), trasferisce il Byte sulle uscite, cambiandone lo stato. Se necessario, è possibile collegare più 4094 in serie tra loro semplicemente collegando sia il clock sia lo strobe di tutti i 4094 in parallelo, l'ingresso 2 del primo shift register al PIC, il piedino 9 (QS) all'ingresso del successivo 4094 (pin 2). In questo modo possono essere collegati in cascata più 4094 e raggiungere le 128/256 o più uscite, limitati dal solo fan-out del PIC. E tutto questo con soli 3 piedini.
Nella **figura 15** potete notare il protocollo di accesso del shift register usato. Il software di gestione del trasferimento del Byte è molto semplice, date un'occhiata alla routine "_tx" per capire quanto sia semplice gestire un shift register. Le routine sono scritte in Assembler ma sono facilmente traducibili in un linguaggio evoluto come il C o il Basic (si veda il **listato 1**). Terminiamo lo schema elettrico con l'analisi del pulsante.
La scelta del pin 4 del PIC non è casuale, perché internamente è collegato al MCLR (Master CLeaR) cioè al reset Hardware del PIC. In fase di caricamento è necessario *non* spuntare questa funzione, permettendo il collegamento interno al PIC come ingresso. Infatti, questo è l'unico pin che può essere configurato unicamente come ingresso ed è l'ideale per testare il pulsante.

MONTAGGIO

Come avrete notato, tutti i componenti trovano posto in due schede distinte, le cui misure e forma si adattano a un contenitore di RETEX, il GIBOX 2. Abbiamo scelto questo contenitore sia per il costo sia per la sua ampia diffusione. Non possiamo fidarci di proporre un'apparecchiatura tanto interessante quanto potenzialmente pericolosa senza vincolarvi al montaggio in un contenitore isolato.
Se optate per i circuiti stampati da noi proposti, vi consigliamo di scaricarli dalla rivista in dimensioni reali. Per la costruzione indirizziamo all'utilizzo dei fogli blu che trasferiscono a caldo il disegno sul supporto semplicemente stampandolo con una stampante laser o una fotocopiatrice.
Non è un sistema particolarmente economico ma per la costruzione di prototipi funziona benissimo, eliminando il tempo di attesa del sistema fotografico e le incertezze per chi usa il pennarello e/o i trasferibili. Se siete organizzati con il sistema della fotoincisione tanto meglio, risultando il più preciso e affidabile, anche se più dispendioso in termini di tempo.
Una volta in possesso dei circuiti stampati, montate prima di tutto i numerosi ponticelli presenti soprattutto nell'area di montaggio dei display.
Proseguite con le resistenze e i diodi, gli zoccoli, i display e tutto il resto. Il pulsante è un modello economico reperibile nei negozi di elettronica.
Sono disponibili tre versioni differenziate dalle dimensioni del piolino di pressione. Nel nostro caso è necessario il modello con il piolino più alto, l'altezza totale è 14 mm (dalla base del CS).
Inserite gli integrati nei propri zoccoli. Il PIC va prima programmato con il file "Ana220V8.HEX" che potete scaricare dal sito della rivista.
Continuate il montaggio con la scheda dell'alimentazione. In questa trova posto il trasformatore di alimentazione, che deve necessariamente essere un modello con due tensioni e lo zero centrale, il ponte raddrizzatore, i condensatori di filtraggio e i regolatori di tensione. In questa scheda è presente la morsettiera per i collegamenti nella linea elettrica e del carico. Questa deve essere adeguata all'utilizzo, perciò scartate modelli troppo piccoli e adatti a tensioni di poche decine di volt.
La scheda dell'alimentazione va montata nelle apposite guide del contenitore, mentre la scheda principale va fissata nel coperchio tramite quattro viti. L'ingombro dei componenti in altezza rende necessario l'utilizzo di quattro distanziali formati ciascuno da un dado da 4 MA e una rondella, per un'altezza totale di 5 mm circa.
Per non trovare difficoltà durante il montaggio, è preferibile incollare i distanziali proposti tramite colla a caldo sul circuito stampato stesso, dal lato componenti.
Prima di fissare il circuito principale al coperchio del contenitore è necessario forarlo in modo da permettere la visualizzazione dei display e dei LED e la fuoriuscita del piolino del pulsante. Anche in questo caso vi proponiamo una possibile maschera. Stampate due copie su supporto autoadesivo, meglio se per uso fotografico.
La prima vi permetterà di forare il coperchio in maniera semplice e precisa, mentre la seconda va applicata definitivamente garantendo un'estetica gradevole. Con un coltellino ben affilato incidete le due diagonali della finestra del display, e ripiegate i 4 triangoli ottenuti verso l'interno.
La carta autoadesiva coprirà eventuali sbavature della finestra del display sempre presenti nelle lavorazioni artigianali. Non forate assolutamente la carta nella zona del pulsante. Quest'ultimo verrà azionato anche se resta sotto il supporto adesivo.
Se per qualsiasi motivo utilizzate come shunt una resistenza autocostruita, vi ricordiamo che la bobina ottenuta va montata perpendicolare allo stampato e lontana dal trasformatore di alimentazione, pena distorsione del segnale della corrente con conseguente errore di visualizzazione ed errato cosϕ.
Prima del montaggio definitivo nel contenitore è necessaria la taratura della scheda principale.

COLLAUDO E TARATURA

Vorremmo prima di tutto focalizzare la vostra attenzione sulla estrema pericolosità della corrente. Fate molta attenzione durante la taratura dello strumento. Usate tutte le precauzioni del caso, se possibile fattevi assistere da una persona che possa intervenire in caso di bisogno. La prima precauzione è il controllo del differenziale che sicuramente avrete nella centralina di casa; effettuate un test prima di iniziare la fase di taratura.
Appurato che il differenziale funzioni e che tolga corrente anche al vostro laboratorio, procuratevi un paio di guanti isolati e prestate molta attenzione a ciò che fate.
Se afferrate i due fili della corrente uno per mano, il differenziale, per quanto efficiente e sensibile (salvavita), non potrà intervenire. L'attenzione non deve abbandonarvi per tutta la durata della taratura.
Una volta montati i due circuiti, alimentate con 5 volt da alimentatore esterno la scheda principale e, se tutto funziona, i display visualizzano 000 e rimane acceso il primo LED che indica la potenza. Se il display non visualizza nulla, controllate che alla locazione di memoria 0x3FF del PIC sia presente il dato nella forma 0x34XX, dove XX rappresenta il valore da inserire nel registro OSCCAL per la calibrazione dell'oscillatore interno. Nel caso il micro sia stato cancellato e la locazione contenga il dato vuoto (0x3FFF), il circuito non potrà funzionare. In tal caso dovrete inserire manualmente un valore da 0x3400 a 0x34FF (0x3480 è il valore centrale).
Appurato il funzionamento, continuiamo la taratura. A ogni pressione del pulsante i LED scorrono dall'alto verso il basso per ricominciare dal primo in un loop continuo.
Al quarto LED, quello che indica il cos , non essendoci segnali da analizzare in ingresso, esso deve indicare 1.00 cioè cosϕ = 1.
Staccate i 5 volt e passate al collaudo della scheda 2. Collegate un cordoncino con un spina da un capo e collegatelo ai morsetti 1 e 2 di X1. Infilate la spina in una presa e controllate con un tester se all'uscita di U2 ci sono +5 volt e su

U3 sono presenti –5 volt. Fatto questo togliete la spina dalla presa dei 220 V, collegate le due schede tra loro prestando attenzione allo schema di **figura 13** e ridate corrente tramite la solita spina.
Premete il pulsante per accendere il secondo LED (*TENSIONE V*), quello che indica la tensione di rete. Con un tester misurate la tensione tra i morsetti 1 e 2 di X1 e tarate V2 fino a leggere lo stesso valore misurato dal tester. Il valore è ammortizzato da C4 e C5, aspettate qualche istante ogni volta che girate il cursore del trimmer.
Togliete la spina e collegate un carico tra i morsetti 3 e 4, sempre di X1, tramite un altro cavetto a cui avrete collegato una presa all'estremità libera. Ridate tensione al circuito e premete il pulsante fino all'accensione del terzo LED (*CORRENTE A*).
In qualche modo dovete conoscere la corrente assorbita dal carico magari collegando, in serie al carico, il tester sulla portata ampère AC adeguata, annotato l'assorbimento regolare V1 per lo stesso valore. Valgono le stesse considerazioni fatte per la tensione.
Valutate le informazioni riguardanti il condensatore C8 riportate nella sezione riservata alla descrizione del cosϕ.
Il circuito tarato sarà pronto all'uso solo dopo averlo sistemato nel suo contenitore completamente isolato.

UTILIZZO

Come avrete intuito, i campi di applicazione del dispositivo sono molteplici, provate ad analizzare il consumo di energia del vostro laboratorio da hobbista o della vostra stanza per scoprire i consumi di energia elettrica anche con tutti gli apparecchi spenti.
Il computer, lo stereo, il televisore e quant'altro possediate, se funziona con un comando a distanza o si accende tramite pulsante consuma – e non poco – se gli apparecchi sono datati. Provate.
All'accensione dello strumento, il display è caricato con il valore della potenza istantanea (*POTENZA KW*).
Ad ogni pressione del pulsante la grandezza visualizzata cambia passando da potenza espressa in x,xx KW alla tensione in volt, quindi la corrente in ampère nella forma xx,x A, nel cos e, infine, nel consumo in KWh per ricominciare da capo.
Non dovete preoccuparvi delle interruzioni di corrente, al ritorno dell'energia elettrica il display si posiziona automaticamente sull'ultimo campo selezionato.
Per non perdere il contenuto del registro dell'energia nelle eventuali interruzioni di corrente, a ogni integrazione, cioè ogni minuto, il registro viene aggiornato e salvato nella EEPROM del PIC.
Nonostante la ditta produttrice del PIC, Microchip, garantisca un numero di riscritture molto elevato nella memoria EEPROM, abbiamo preferito implementare una routine che utilizza una parte di memoria dinamica allungando di fatto la vita della EEPROM di venti volte. La visualizzazione dell'energia prelevata avviene in modo dinamico, per apprezzarne anche i consumi dei piccoli carichi.
La forma iniziale di visualizzazione sarà x.xx KWh, superati i 9,99 KWh la forma visualizzata diventa xx,x KWh perdendo in definizione ma dopo i 10 KWh i centesimi di KWh contano poco. Infine nel caso l'energia superi i 99,9 KWh sono mostrati unicamente i KWh nella forma xxx KWh.
Al fine di un utilizzo vario, per testare e provare sempre nuove esperienze, il registro dell'energia deve essere resettabile. Per cancellare il contenuto del registro è sufficiente premere il pulsante fino alla visualizzazione del cos : ripremete e tenete premuto il pulsante. Il display lampeggia sul vecchio valore per circa 6 secondi infine mostra tutti 0 confermando il reset del contatore.
Le **figure 8**, **9**, **10**, **11** e **12** mostrano i valori rilevati collegando un tostapane da 500 W. ❑

CODICE MIP 2805111

CODICE MIP 2798032

a cura della redazione

Un webinar *gratuito* per usare l'oscilloscopio

Una lezione gratuita direttamente da LeCroy sull'uso degli oscilloscopi

L'oscilloscopio digitale è uno degli strumenti di test più comunemente usati sul banco di misura. Le potenzialità di misurazione sono molteplici: misure in continua, in alternata, rilevazione dei transitori… il segreto è saper usare a fondo questo strumento.

LeCroy

LeCroy, in collaborazione con Fare Elettronica, presenta il webinar gratuito: ***Oscilloscope Basics - Ottieni il massimo dal tuo Oscilloscopio Digitale***

Questo webinar gratuito ti permetterà di capire come è fatto e come funziona un oscilloscopio per poterlo utilizzare al meglio. Puoi registrarti gratuitamente all'indirizzo www.farelettronica.com/webinar e seguire comodamente da casa tua la lezione che sarà tenuta dall'Ing. Tuveri di LeCroy con la moderazione dell'ing. Maurizio Del Corso di Fare Elettronica. Il webinar si terrà il giorno ***21 novembre 2011*** *alle* ***ore 14,30*** *e sarà in lingua italiana. Tutto ciò di cui hai bisogno è un PC con una connessione ad Internet.*

Ti aspettiamo!

elettro quiz 317

rispondi & vinci

*Le risposte ai quiz "Base" e "Avanzato" vanno inviate esclusivamente compilando il modulo su www.farelettronica.com/eq specificando la parola chiave **"Tesla"**.*

Le risposte ed i vincitori (previa autorizzazione) sono pubblicati alla pagina www.farelettronica.com/eq a partire dal 15 del mese successivo alla pubblicazione sulla rivista.

A tutti i partecipanti verrà assegnato un buono sconto del 10% (validità 3 mesi dalla data di assegnazione) utilizzabile per un prossimo acquisto su www.ieshop.it

base

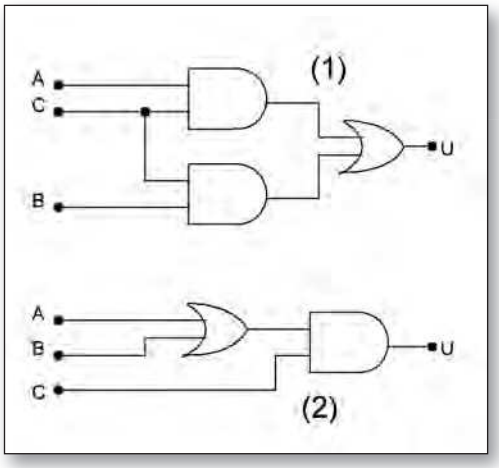
In relazione ai due circuiti proposti in figura, quali delle seguenti affermazioni sono vere?

A) Se A=1 il circuito (1) è un AND tra B e C

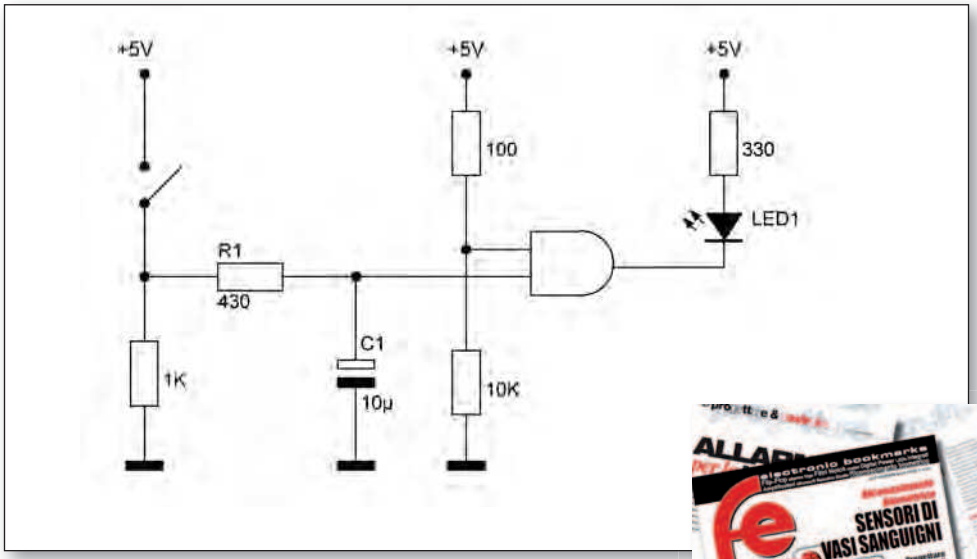
B) I due circuiti svolgono la medesima funzione

C) Se C=0 l'uscita dei due circuiti è 0 indipendentemente dagli altri ingressi

D) il circuito (1) è equivalente ad una porta XOR



Se rispondi correttamente potrai vincere l'abbonamento (o rinnovo) al Club di Fare Elettronica per 1 anno.



avanzato

Nel circuito di figura la porta AND è di tipo CMOS con tensione di soglia 2,5V, ovvero l'ingresso è riconosciuto alto se maggiore di 2,5V, altrimenti è riconosciuto come livello logico basso. Si determini lo stato del LED quando l'interruttore è aperto e nei 10 secondi successivi alla chiusura.

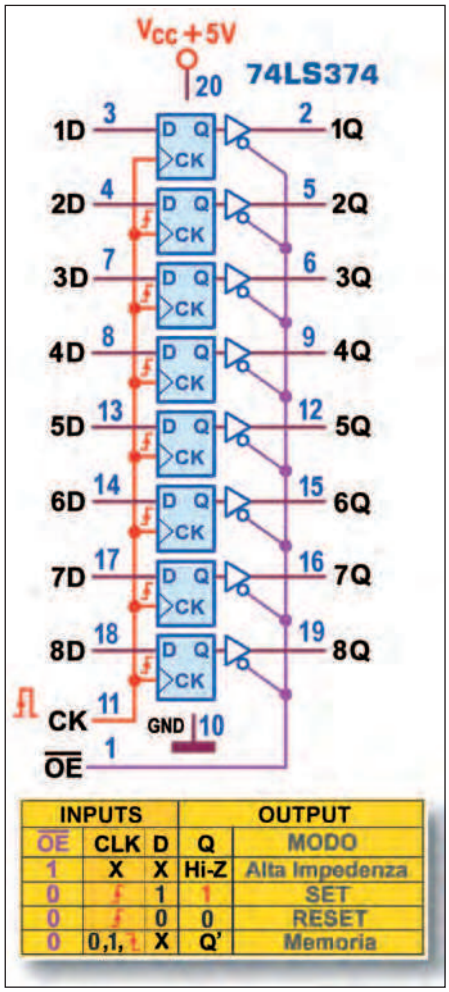
Per i più bravi in palio l'abbonamento annuale a Fare Elettronica.

Interfacciamento dei processori *(parte prima)*

Interfaccia d'uscita PER DATI *a 8 bit*

La grande richiesta di modernità e di efficienza ha praticamente ridotto all'osso il prezzo dei processori e dei componenti affini: la moderna tecnologia non ne può più fare a meno ed essi sono ormai presenti in ogni dispositivo; vediamo insieme qualche idea per utilizzarli al meglio

L'utilizzo dei processori è una pratica ormai inderogabile: nessuno degli apparecchi più comuni può farne a meno, dai telefonini ai navigatori satellitari, dai tablet ai lettori mp3, dalla lavatrice di casa all'impianto di riscaldamento ecc. In realtà questi oggetti sono veri e propri microcomputer: il *microprocessore* (o, sinteticamente, il *processore*, la *CPU* – Central Processing Unit, unità centrale di processo) è solo una parte dei dispositivi programmabili che li governano, per altro quella più importante, chiamata a controllare e decidere, a eseguire ed elaborare. Ma nulla potrebbe questo potente cervello se non disponesse della memoria e delle porte di ingresso e di uscita: a) la *memoria* è il luogo nel quale il processore trova conforto: la sua grande capacità, le sue enormi potenzialità sono niente senza la memoria; la prima cosa che fa, senza saperlo e senza volerlo, per istinto, è saltare in una locazione di memoria, alla ricerca di qualche cosa da fare...Dunque, la memoria è indispensabile in un computer perché contiene il programma che il processore è chiamato a eseguire e perché si presta, eventualmente, a conservare i dati da esso elaborati; b) nessun controllo di dispositivo potrà essere esercitato dalla CPU se non dispone di una struttura di inter-



INPUTS			OUTPUT	
$\overline{OE}$	CLK	D	Q	MODO
1	X	X	Hi-Z	Alta Impedenza
0	↑	1	1	SET
0	↑	0	0	RESET
0	0,1, ↓	X	Q'	Memoria

Figura 2: 3-State Octal D-Type Flip-Flops 74LS374: schema funzionale e Tabella di Verità.

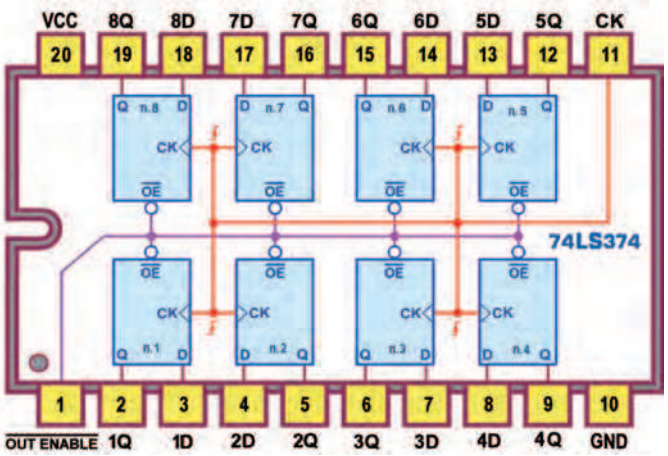


Figura 1: 3-State Octal D-Type Flip-Flops 74LS374: pin-out.

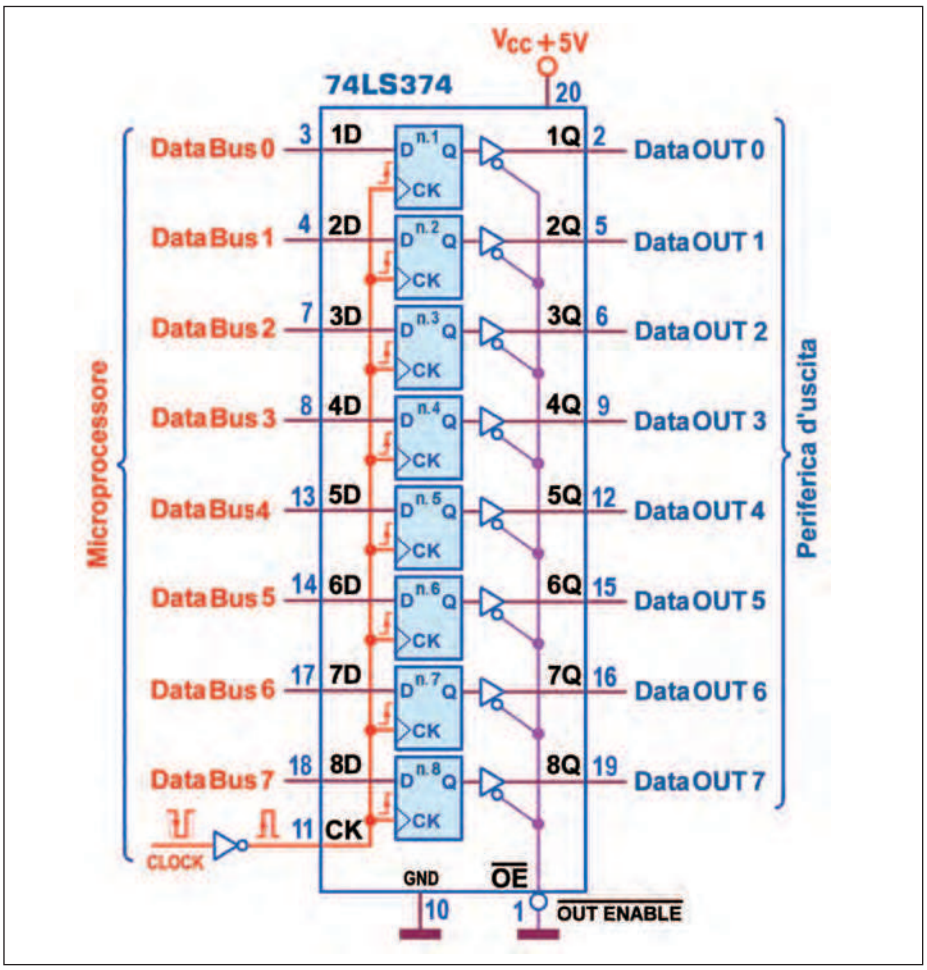


Figura 3: 3-State Octal D-Type Flip-Flops 74LS374: porta (interfaccia) d'uscita.

scambio, uno o più canali delegati alla trasmissione e alla ricezione delle informazioni con il resto del mondo. In sintesi, per funzionare, qualunque computer deve disporre di tutti e tre questi componenti e, per essere economici e competitivi, da molto tempo i vari produttori hanno provveduto a integrarle in un unico dispositivo, noto come *single-chip* o *PIC* (Peripheral Interface Controller o Programmable Intelligent Computer) o *microcontrollore*. Poiché l'appetito vien mangiando, i moderni microcomputer, oltre alla memoria *RAM* (per i dati) e quella *flash* (per il programma) ospitano internamente numerose *porte parallele* (programmabili sia in uscita sia in ingresso) e almeno una *porta seriale RS232*, completando la dotazione con alcuni *temporizzatori* (timer), *comparatori* e *convertitori analogico digitali* (ADC): un vero lusso!

Naturalmente il controllo dei dispositivi esterni può essere esercitato anche con un computer tradizionale, scrivendo programmi in grado di gestire le sue porte classiche, parallela e seriale; l'avvento delle nuove interfacce seriali (USB, Firewire ecc.) rende comunque questo esercizio sempre più impegnativo.

Sono numerosi gli appassionati che si lasciano tentare da questa nuova frontiera, affidando la loro volontà e le loro speranze a un *ATMEL* o a un *MicroChip* o a un *Stmicroelectronics* ecc.: una enorme disponibilità di single-chip, tutti con la loro architettura, la loro dignità e il loro dialetto; oggetti operativamente molto simili tra loro, ma sostanzialmente differenti. L'utilizzo pratico dei microcontrollori non è difficile e non deve intimorire: a) tutti hanno bisogno di una serie di *registri* (la memoria personale interna con la quale rendono possibili le operazioni logiche e aritmetiche, tipiche di ciascuno di essi); poiché essi sono inseriti in una *architettura* tipica, è necessario conoscerla bene per poterli utilizzare al meglio; b) tutti hanno bisogno di un *set d'istruzioni*: esso cambia effettivamente da processore a processore ma è sostanzialmente lo stesso su ogni piattaforma e differisce solo per lo mnemonico usato (ad esempio, per spostare dati, taluni processori utilizzano *MOV* invece di *LD* o di *LOAD* o altro): le varie "razze" di processori parlano la stessa lingua ma con dialetti diversi! La programmazione dei microcontrollori è dunque un'arte che va consumata con pazienza e amore; il modo più immediato ed efficiente fa riferimento all'Assembly (come detto, tipico di ciascuno di loro) anche se è sempre più diffusa la possibilità di fruire di ambienti di programmazione ad alto livello (di solito C-language, ma anche Pascal o altro). L'utilizzo di linguaggi evoluti, decisamente più vicini all'uomo e meno alla macchina (cioè lo stesso single-chip) è un'arma a doppio taglio: il codice così prodotto è certamente molto più lungo, richiedendo maggiore spazio per il suo stoccaggio sulla memoria flash; il problema è però sempre meno preoccupante perché con il passare del tempo la dimensione della memoria di programma garantita on-board è sempre più grande. Il problema eventualmente da valutare con attenzione è legato ai tempi di esecuzione di un programma progettato in C o in Pascal, certamente di gran lunga maggiori di quelli ottenibili con un codice generato con l'Assembly; ma chi non ha tempo da perdere trova fantastico delegare compiti importanti (come la gestione di un visualizzatore LCD o il controllo di una periferica seriale) a procedure già pronte. Chi può dar loro torto? con buona pace di coloro (ahimè, come il sottoscritto) che amano conoscere il perché delle cose.

Ma nel nostro mondo qualcuno ha deciso per noi che bisogna correre, produrre e trarre il massimo profitto, subito e possibilmente gratis, senza spendere nulla di proprio, sfruttando l'intelligenza altrui e mortificando la propria.

L'INTERFACCIAMENTO

Per entrare nel merito degli argomenti di questa serie di articoli, qualunque computer (microcontrollore o personal) si propone di garantire il proprio servizio alle cosiddette *periferiche* (stampanti, mouse, tastiere, visualizzatori, monitor, hard disk ecc.) e per questo deve disporre di *interfacce*, cioè di uno strato di elettronica in grado di mediare tra le offerte del primo con le necessità del secondo. Il compito più banale di una interfaccia è quello di assicurare *da entrambe le parti* i segnali elettrici adatti per rendere possibile lo scambio di informazioni tra di esse: è noto a tutti che l'informazione (sinteticamente *bit*, binary unit, organizzata comunque in *byte*, cioè a gruppi di 8 bit) altro non è che un livello di tensione presente (= "1") o assente (="0") su un filo conduttore. Tuttavia una interfaccia può anche essere, a sua volta, intelligente e specializzata, con l'aiuto di componenti sofisticati (come l'*UART* della seriale) o funzionalmente indispensabili (come *buffer di memoria* o *buffer di corrente*). Faccio notare come, nel comune gergo dell'elettronica digitale, il termine *buffer* sia utilizzato: a) sia per indicare un pugno di bit disponibili a mantenere (memorizzare) localmente una certa quantità di livelli logici a disposizione all'occorrenza (per esempio per mantenere acceso un digit a sette segmenti mentre il controllore è occupato con altri compiti); b) sia per indicare uno strato di elettronica in grado di garantire la corrente necessaria a una periferica, cosa decisamente impossibile per qualunque porta d'uscita del micro. In queste nostre considerazioni ci occuperemo ora della prima tipologia: un buffer di memoria a 8 bit, cioè di un circuito riconoscibile come *interfaccia d'uscita* a 8 bit.

INTERFACCIA D'USCITA A 8 BIT

I dispositivi di controllo programmabili sono sempre dotati di porte adatte a trasferire (in modo bidirezionale) informazioni di tipo parallelo ad eventuali periferiche; i microcontrollori dispongono di più di una porta, di norma a 8 bit, mentre un personal computer dispone (ahimè, sempre con meno frequenza) di una sola porta a 8 bit, tipicamente nota come *porta parallela*. In entrambi i casi ciascuna delle 8 linee può essere programmata per prendere (*IN*) o dare (*OUT*) un'unità di informazione (*bit*), da scambiare in blocco (cioè sotto forma di *byte*) con i registri interni della CPU (di solito della stessa dimensione) al fine di elaborarne velocemente il valore, in modo aritmetico o logico. In questa puntata ci occuperemo del caso in cui un *byte* viene fatto uscire dal Bus Dati; il processore che governa la porta d'uscita (nell'ambito del PC o del single-chip) sarà chiamato a eseguire un paio di istruzioni (in *Assembly*), sufficienti per imprimere in tempo reale (qualche nanosecondo) un ben preciso valore binario su ciascuna delle sue 8 linee, lasciandolo a disposizione fino a quando deciderà di sostituirlo con un altro, continuando, subito dopo, ad occuparsi di altre attività. In questa situazione, la presenza di hardware (interfaccia) aggiuntivo esterno potrebbe sembrare non necessaria, ma è comunque fortemente consigliata, per garantire una adeguata corrente di pilotaggio (raramente disponibile in modo diretto sui canali relativamente delicati dei dispositivi di controllo) o, semplicemente, per il fatto che un guasto (un corto!) su di essi può determinare la crisi di tutto il microprocessore, se non di tutto il computer che lo ospita. Per assolvere a questo compito, il componente più versatile è il 74LS374, ottuplo flip-flop D-Type con uscite 3-state; il suo pin-out è illustrato in **figura 1**. Esso contiene 8 elementi di memoria (*flip-flop*) di tipo *D-*

TABELLA 1

CARATTERISTICHE 74LS374	SIMBOLO	VALORI
potenza dissipata massima (con ingressi a "0" e con uscite in Hi-Z)	ICC * VCC	225 mW =45 mA * 5V
corrente erogata massima in uscita	I_{OH}	2,6 mA (con uscita VOH a "1")
corrente assorbita massima in uscita	I_{OL}	24 mA (con uscita VOL a "0")
ritardo di propagazione massimo da dato a uscita	t_{PLH}	18 ns
	t_{PHL}	18 ns
ritardo di propagazione massimo da clock e da enable a uscita	t_{PLH}	28 ns
	t_{PHL}	28 ns
tempo di abilitazione d'uscita massimo	t_{PZH}	28 ns
	t_{PZL}	28 ns
tempo di disabilitazione d'uscita massimo	t_{PHZ}	20 ns
	t_{PLZ}	25 ns
massima frequenza di clock	f_{MAX}	35 MHz (minima), 50 MHz (tipica)
setup time minimo	t_S	20 ns
hold time minimo	t_H	1 ns

Tabella 1: 3-State Octal D-Type Flip-Flops 74LS374: caratteristiche elettriche e dinamiche.

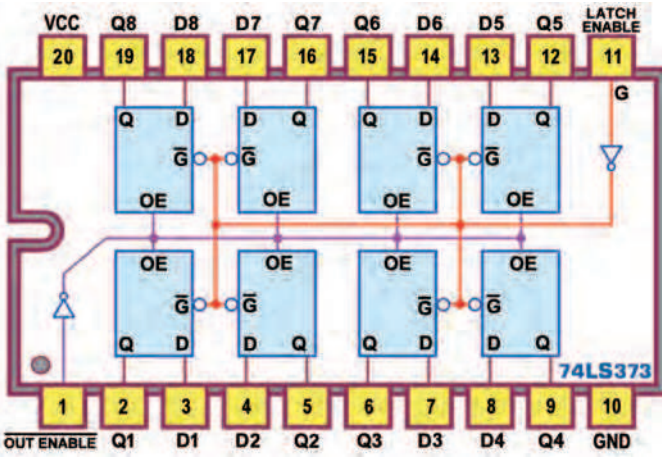


Figura 4: 3-State Octal Transparent D-Latches 74373: pin-out.

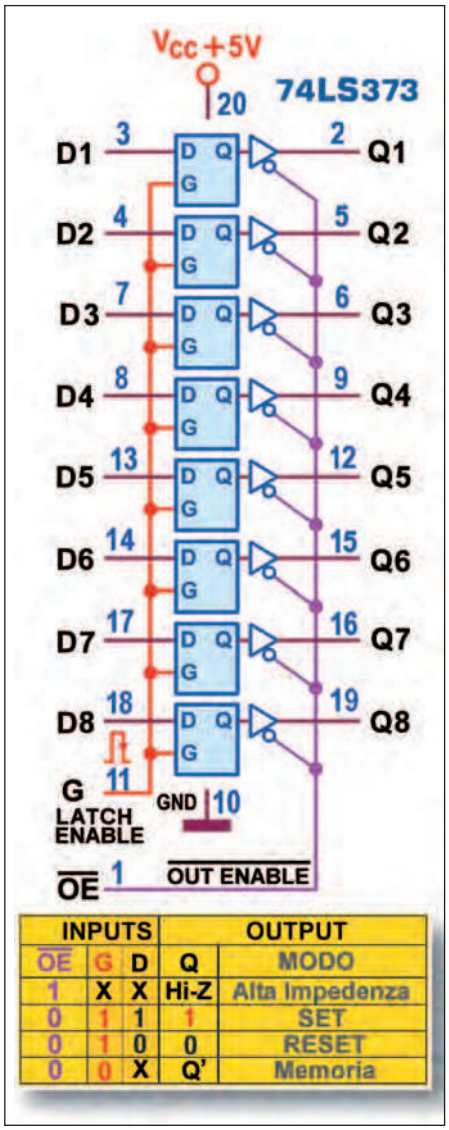


Figura 5: 3-State Octal Transparent D-Latches 74373: schema funzionale e Tabella di Verità.

Type, caratterizzati dal fatto di agire contemporaneamente sul fronte della linea di comando (pin 11, CK, *Clock*): a) quando il segnale di controllo passa dal livello basso 0 a quello alto 1 (fronte di salita) l'uscita Q tiene (memorizza) il valore logico predisposto in precedenza sull'ingresso corrispondente D; proprio per questo suo comportamento i manuali definiscono *Positive Edge-Triggered Flip-Flops* gli elementi di questo componente; b) in ogni altro istante della forma d'onda del clock CK su ciascuna uscita Q rimane disponibile il valore logico frutto della precedente operazione di memorizzazione: non sarà possibile imporre cambiamenti alle uscite, per cui esse saranno del tutto indifferenti a ogni variazione dei rispettivi ingressi. I *D-Type* di questo componente sono del tutto uguali agli elementi di memoria che costituiscono la *cache* del nostro computer, cioè di *Static RAM*, *SRAM*; osservando il dettaglio funzionale del componente **(figura 2)** si capisce perché la SRAM sia più costosa e più veloce della *DRAM* (*Dynamic RAM*) di solito usata per la memoria convenzionale: ogni suo bit deve essere infatti realizzato con un piccolo, ingombrante, circuito elettronico bistabile (appunto flip-flop) mentre quello della DRAM è costituito da un semplice condensatore allo stato solido, controllato da un solo transistore MOS, in grado di memorizzare un 1 logico (se è carico) o uno 0 (se è scarico); sebbene sia necessario un circuito di *refresh*, per "ricaricare" a intervalli regolari di qualche nanosecondo i condensatori a 1 di un'intera riga, la dimensione ridotta delle "celle di memoria" rende insostituibile questa tecnologia per le memorie di grande capacità. In aggiunta alla sua tipica funzione di memoria il componente offre il servizio di *buffer di corrente*, cioè dispone di uscite amplificate non invertenti di tipo 3-state, controllate contemporaneamente da un ulteriore piedino di abilitazione (pin 1, OE, Out Enable), attivo basso; la funzione di OE è del tutto simile a quella esercitata dalle abilitazioni Gx del componente TTL 74LS244 (esplicitamente e solo buffer): a) quando è attiva (cioè a massa, OE=0) consente il passaggio in uscita del livello logico presente sull'uscita interna del flip-flop; b) quando non è attiva (cioè scollegata o posta a 1) mette in alta impedenza le 8 uscite esterne del componente; il valore presente sulle uscite dei rispettivi flip-flop rimane così congelato. È importante capire che cosa significa mettere un'uscita *in alta impedenza*; supponiamo di collegare una tensione ai capi di un resistore: a) la corrente che lo attraversa è soggetta alla legge di Ohm per cui, a parità di tensione applicata, più grande è la sua resistenza e minore è la corrente che passa; b) se la resistenza elettrica è infinita la corrente è nulla, cioè un resistore con questa caratteristica si comporta come un *circuito aperto*, come se esso stesso fosse scomparso, lasciando scollegati i punti ai quali era collegato. Dunque, la condizione di alta impedenza delle uscite *scollega* praticamente l'oggetto a esse collegato, anche se fisicamente la cosa non si vede! Tecnicamente si dice che le uscite di questo componente hanno 3 stati logici: i consueti 1 e 0 e il terzo stato, detto *Hi-Z,* appunto "alta impedenza".

La possibilità di forzare le uscite in alta impedenza si rileva fondamentale e indispensabile nel progetto delle logiche programmabili che richiedono strutture organizzate a *bus*; possiamo immaginare due o più integrati come quello di cui stiamo parlando ora, tutti con le uscite di pari peso collegate tra loro (per esempio la 1Q di uno con la 1Q di tutti gli altri, *in parallelo*): è chiaro che solo un componente alla volta può postare il proprio livello logico sul bus. Di solito se si collegano tra loro le uscite di due oggetti logici si commette un errore fatale: se (nella migliore delle ipotesi) avessero lo stesso valore logico andrebbe ancora bene, ma se i li-

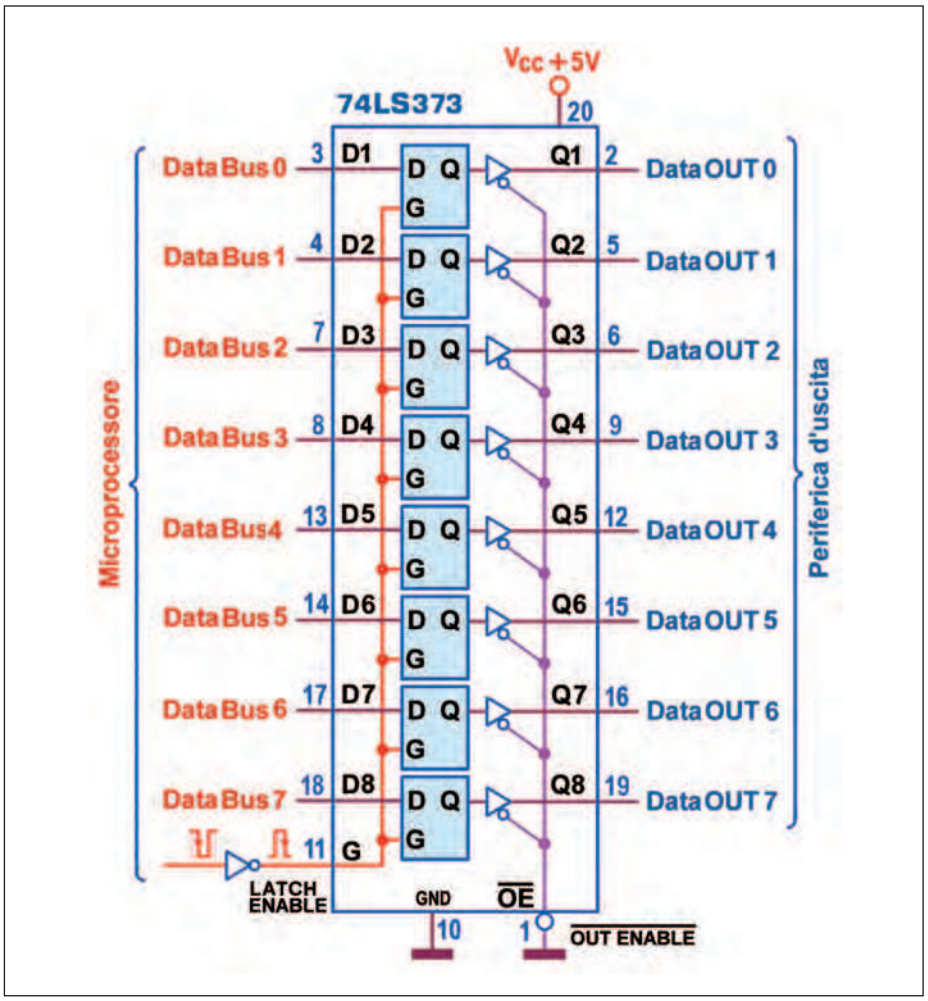


Figura 6: 3-State Octal Transparent D-Latches 74373: porta (interfaccia) d'uscita.

velli presenti sulle due uscite fossero alternativi, una delle due (quella a livello 0) provocherebbe il cortocircuito dell'altra, forzandola a massa con il suo 0, con effetti estremamente negativi; ma se entrambe fossero di tipo 3-state, ponendone una in Hi-Z il problema non esisterebbe, dato che nessun livello logico può essere influenzato da un circuito aperto.

La possibilità di disporre dell'alta impedenza sulle uscite ci suggerisce un'altra importante situazione: essa è utile nelle strutture organizzate a bus (tipiche del progetto con logiche programmabili): per esempio è possibile collegare più memorie 74LS374 in parallelo, attivandone all'occorrenza solo una, alternativa alle altre. Nel normale singolo utilizzo, la possibilità di disporre dell'alta impedenza delle uscite non è necessaria, per cui OE si collega fisso a massa (il che assicura comunque la gradita azione di *buffer di corrente* su ciascuna linea d'uscita); il componente utilizzato in questo modo **(figura 3)** si comporta come *porta d'uscita*, e il suo impiego è particolarmente consigliato per "mettere in sicurezza" quelle dei microprocessori o dei single-chip; la funzionalità interna dei flip-flop viene comunque garantita anche con OE=1: mentre le uscite sono in alta impedenza i vecchi dati possono essere mantenuti oppure sostituiti con quelli nuovi, eventualmente in arrivo dall'esterno.

La porta d'uscità è dunque pronta per gestire in blocco un'informazione di 8 bit (un byte) in transito sul *bus dati* della CPU, bloccandola in un ben preciso istante, a disposizione della *periferica*, per esempio una batteria di 8 LED o di 8 relé, o un digit a 7 segmenti, e così via.

L'analisi di questo circuito ci consente di evidenziare un aspetto spesso sottovalutato: il sincronismo (clock) disponibile per la memorizzazione è quasi sempre un segnale tenuto alto per quasi tutta la sua esistenza funzionale; solo nel momento desiderato per la memorizzazione viene generato un brevissimo impulso basso; ricordando che la natura *D-Type* dei singoli flip-flop del 74LS374 li rende attivi solo su uno dei due fronti, va sottolineato che: a) la memorizzazione avviene comunque *sul fronte di salita*; b) in nessun altro istante dell'impulso sarà possibile imporre cambiamenti alle uscite, per cui esse sono del tutto indifferenti a ogni variazione dei rispettivi ingressi, e la durata dell'impulso è del tutto irrilevante; c) poiché tipicamente l'impulso di sincronismo è attivo basso, se esso fosse fornito direttamente sul pin 11 di clock (CK) l'effettiva memorizzazione avverrebbe con un ritardo pari alla durata dell'impulso fornito; d) è consigliabile dunque inserire un inverter sul citato pin di clock, per garantire la memorizzazione esattamente nell'istante in cui il sincronismo va basso (trascurando i 20 ns del ritardo di propagazione), ma, se la durata dell'impulso è effettivamente breve, la presenza dell'inverter esterno può essere ritenuta un "lusso" inutile. La **tabella 1** raccoglie le principali caratteristiche elettriche e dinamiche del componente (i tempi sono stati rilevati con carico di 667ohm/45pF).

Un'interessante alternativa al 74LS374 è offerta dal componente 74LS373, ottuplo D-Latch con uscite 3-state; la **figura 4** mostra il suo pin-out, compatibile con quello del precedente, salvo che per la natura del segnale presente sul pin 11, utilizzato per memorizzare le informazioni proposte sulle sue linee d'ingresso. Gli 8 elementi di memoria contenuti nell'integrato sono ora di tipo *D-Latch*, limitati dall'effetto "trasparenza", tipico di questa categoria, e caratterizzati dal fatto di agire non più sul fronte attivo (*edge triggered*) di un *clock* ma sul livello (*level triggered*) di una linea di abilitazione, *enable*, G: a) quando G=1, scollegato o fissato al positivo dell'alimentazione, il dato D passa inalterato sull'uscita corrispondente Q, in modo completamente asincrono: ciascun elemento di memoria si comporta come una porta aperta (l'uscita insegue l'ingresso), cosicché si dice essere *trasparente* (come se non ci fosse!); proprio per questo suo comportamento i

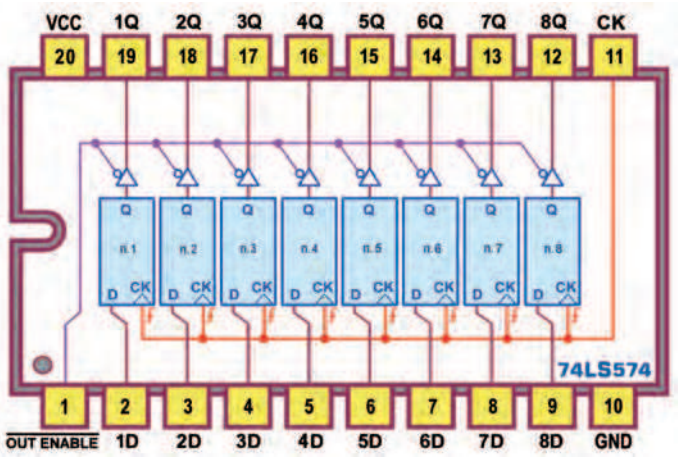


Figura 7: 3-State Octal D-Type Flip-Flops 74LS574: pin-out.

manuali definiscono questo componente *Octal Transparent Latches*; b) quando G è forzata a massa, 0, l'uscita Q tiene (memorizza) il valore logico che aveva il corrispondente ingresso D nel momento del passaggio da un livello (alto) all'altro (basso), cioè sul fronte di discesa.
ID-Latch di questo componente, con un'immagine figurata, si possono dunque ritenere *porte con memoria*: con l'abilitazione alta, sulle uscite passa ogni variazione degli ingressi, fino all'arrivo di un fronte di discesa, in corrispondenza del quale l'ultimo dato presente viene mantenuto in uscita per tutto il tempo in cui l'abilitazione rimane bassa. La **figura 5** riassume quanto detto nella Tabella di Verità e offre l'occasione per confrontare funzionalmente i due componenti, al fine di operare una scelta oculata nel caso si desideri utilizzarli come porta d'uscita di un sistema programmabile; in particolare (come per il 74LS374) è evidente il servizio di *buffer di corrente*, essendo ciascuna uscita dotata di amplificatori non invertenti di tipo 3-state, progettati specificamente per pilotare carichi con impedenza relativamente bassa e controllati contemporaneamente da un ulteriore piedino di abilitazione (Out Enable, OE), attivo basso; valgono le stesse considerazioni di prima: quando è attiva (OE=0) consente il passaggio (sull'uscita effettiva) del livello logico presente sull'uscita interna del Latch, mentre quando non è attiva (OE=1) le 8 uscite esterne sono forzate in alta impedenza, isolandole di fatto da qualunque dispositivo a valle ad esse collegato. La **figura 6** mostra la tipica applicazione come *porta d'uscita* (analoga a quella già proposta per il 74LS374), cioè l'interfaccia tra gli 8 ingressi di un utilizzatore e (presumibilmente) le linee del bus dati di un microprocessore o di una porta di un single-chip o le linee d'uscita della porta parallela di un PC; anche in questo caso la necessità di disporre dell'alta impedenza sulle uscite non è necessaria, per cui OE va collegato a massa, senza per questo vanificare la loro capacità di *buffer di corrente*. È importante e istruttivo confrontarne l'aspetto funzionale con quello del 74LS374, sempre nell'ipotesi di utilizzare, per la memorizzazione, un segnale di sincronismo tipicamente tenuto quasi sempre alto, meno che nel momento desiderato per la memorizzazione, durante il quale viene portato basso per un brevissimo istante; ricordando che la natura *D-Latch* dei singoli elementi di memoria del

TABELLA 2

CARATTERISTICHE 74LS373	SIMBOLO	VALORI
potenza dissipata massima	ICC * VCC	200 mW =40 mA * 5V (con ingressi a "0" e con uscite in Hi-Z)
corrente erogata massima in uscita	I_{OH}	1 mA (con uscita VOH a "1")
corrente assorbita massima in uscita	I_{OL}	12 mA (con uscita VOL a "0")
ritardo di propagazione massimo da dato a uscita	t_{PLH}	18 ns
	t_{PHL}	18 ns
ritardo di propagazione massimo da clock e da enable a uscita	t_{PLH}	30 ns
	t_{PHL}	30 ns
tempo di abilitazione d'uscita massimo	t_{PZH}	28 ns
	t_{PZL}	36 ns
tempo di disabilitazione d'uscita massimo	t_{PHZ}	20 ns
	t_{PLZ}	25 ns
massima frequenza di clock	f_{MAX}	35 MHz (minima), 50 MHz (tipica)
setup time minimo	t_S	5 ns
hold time minimo	t_H	20 ns

Tabella 2: 3-State Octal Transparent D-Latches 74LS373: caratteristiche elettriche e dinamiche.

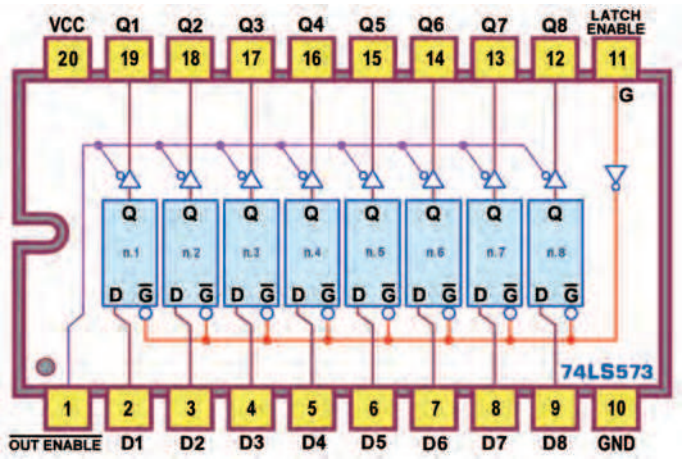


Figura 8: 3-State Octal Transparent D-Latches 74573: pin-out.

74LS373 li rende attivi sui livelli, in questo caso va sottolineato che: a) la memorizzazione del byte avviene comunque *sul fronte di discesa*; b) per tutto il tempo in cui il segnale rimane alto le uscite sono *trasparenti*, cioè fortemente sensibili a ogni variazione dei rispettivi ingressi, per cui "così com'è" il sincronismo non va bene: è assolutamente indispensabile trasformarlo in un impulso alto e breve, e assicurarsi che i dati in ingresso siano stabili, almeno per un tempo un po' più grande dell'impulso positivo (cioè siano forniti poco prima e tolti poco dopo la sua presenza); c) per entrambe le precedenti ragioni, con impulsi di sincronismo tipici (attivi bassi) per un corretto funzionamento del componente è necessario inserire un inverter direttamente sul pin 11 di Enable (G); d) la presenza dell'inverter esterno ritarda comunque l'effettiva memorizzazione di un tempo pari alla durata (di norma trascurabile) dell'impulso fornito, in aggiunta al suo tempo di propagazione (20 ns). La **tabella 2** raccoglie le principali caratteristiche elettriche e dinamiche del componente (i tempi sono stati rilevati con carico di 667ohm/45pF).

CONSIDERAZIONI FINALI

Deve essere ben chiaro che le interfacce proposte sono in grado di garantire il servizio di *porta d'uscita* in modo completamente autonomo e autosufficiente, a partire direttamente dal bus dati di un processore; ciò non toglie che il loro impiego (pur per certi versi ridondante) possa tornar utile anche a valle delle porte intrinsecamente già disponibili in un microcontrollore (nei quali peraltro il bus dati della CPU non è accessibile), per motivi di sicurezza e/o di bufferizzazione; in questo secondo caso (non essendo particolarmente necessaria la funzione di memoria) potrebbe essere sufficiente l'impiego di componenti come gli *Octal Buffer/Line Driver with 3-State Outputs*, 724LS244 (non invertente) e/o 74LS240 (invertente), dei quali ci occuperemo in altra occasione. In ogni caso, se si opta per l'utilizzo del 74LS374 o del 74LS373, l'attenta lettura dei due schemi proposti come *interfaccia* mette in evidenza il fatto che la CPU dovrà fornire (in aggiunta alle 8 necessarie per assicurare il dato alla periferica) almeno un'ulteriore linea di uscita, da fornire sul pin 11 di entrambi i componenti per garantire il segnale di sincronismo necessario alla memorizzazione del dato, intesa come Clock (CK) per i *D-Type* del 74LS374 e come Enable (G) per i *D-Latch* del 74LS373. Essa dovrà essere resa disponibile utilizzando un bit di una seconda porta (di un single-chip) o uno dei segnali di controllo (della porta parallela di un PC); *modulando* da software il valore logico presente su questa linea dovrà essere ricostruita la forma d'onda (discussa e descritta in precedenza) necessaria per assicurare il comando di scrittura, in funzione del componente utilizzato per la memorizzazione.

In particolare il programma dovrà provvedere a tenere questa linea a livello basso (0 logico) fino al momento in cui è richiesta la memorizzazione, nel quale verrà portata a 1 logico per un istante molto breve, dopo il quale sarà forzata nuovamente a 0; in questo modo è possibile garantire la memorizzazione sia con il *D-Type* 74LS374 che con il *D-Latch* 74LS373, perché la durata estremamente breve dell'impulso positivo così prodotto farà di fatto coincidere il fronte di salita (necessario per il primo) con il fronte di discesa (necessario al secondo). La gestione da software del segnale di sincronismo rende quindi inutile la presenza dell'inverter sul pin 11 di entrambi gli integrati, necessaria invece nella gestione da hardware per via del fatto che, in questo àmbito, i segnali disponibili per questo scopo sono *nativamente* impulsi attivi bassi: si pensi per esempio alle uscite di un decoder binario, molto usato in questi casi. Per finire voglio ricordare altri sei componenti, altrettanto adatti per lo scopo trattato in questa nostra prima occasione; i primi due hanno la stessa piedinatura e le stesse caratteristiche elettriche e dinamiche di quelli descritti ma restituiscono uscite 3-state invertenti; sono il 74LS534 (*Octal D-Type Flip-Flops* per il quale valgono le **figure 1**, **2** e **3**, tenendo conto che ora le uscite sono negate) e il 74LS533 (*Octal Transparent D-Latches* per il quale valgono le **figure 4**, **5** e **6**, sempre con uscite negate). Osservando i componenti descritti in precedenza salta subito agli occhi la strana collocazione scelta dal costruttore per la disposizione degli ingressi D e delle uscite Q, posti a coppie dallo stesso lato del circuito integrato; questo fatto può aggiungere un disagio in più al progettista del circuito stampato chiamato ad ospitarli; probabilmente per questa ragione sono stati resi disponibili i componenti 74LS574 (*D-Type*) e 74LS573 (*D-Latch*), funzionalmente identici rispettivamente al 74LS374 e 74LS373, ma con tutti gli 8 ingressi sul lato sinistro del chip e tutte le 8 uscite sul lato destro; il pin-out del primo è disponibile in **figura 7** e quello del secondo in **figura 8**. Infine, anche per la coppia (appena presentata) di componenti con tutti gli 8 ingressi sul lato sinistro del chip e tutte le 8 uscite sul lato destro esiste una versione con uscite 3-state invertenti; sono il 74LS564 (*D-Type*) e il 74LS563 (*D-Latch*), per il pin-out dei quali possiamo far riferimento ancora (rispettivamente) alla **figura 7** e alla **figura 8**, tenendo conto che ora le uscite vanno intese negate. ❑

CODICE MIP 2814143

entra in un mondo di contenuti digitali

CLUB DELLE RIVISTE

E-BOOK

ANNATE IN PDF

fare elettronica DIGITAL EXTRA

il riferimento per tutta l'elettronica sul tuo PC o TABLET

E-BOOK

I migliori libri di elettronica in formato e-book.
Puoi leggere e stampare i capitoli o le pagine di interesse (ad alta risoluzione) e installare i software eventualmente trattati nei volumi.

ANNATE IN PDF

Le annate complete delle riviste Fare Elettronica, Firmware ed Elektor in comodi file zip navigabili o masterizzabili su CD o DVD completi di tutte le riviste (e relativi downloads) uscite nei vari anni, da non perdere!

CLUB DELLE RIVISTE

Scarica illimitatamente per un anno centinaia di articoli pubblicati sulle riviste Fare Elettronica e Firmware.

fare elettronica DIGITAL EXTRA

Monografie di elettronica in formato digitale: raccolte di progetti completi, CNC e microcontrollori.

www.ieshop.it

INWARE EDIZIONI

CODICE MIP 2814086

l'angolo di Mr A.KEER

SHIFT REGISTER

(seconda parte)

La presenza dei Flip-Flops D-Type nell'àmbito dei Registri a Scorrimento ribadisce la loro straordinaria utilità; dopo l'analisi dettagliata della modalità SISO vediamo ora quella SIPO, una variante della precedente che offre nuovi interessanti orizzonti applicativi

La categoria dei Registri a scorrimento (Shift Registers) propone una grande quantità di componenti basati sui Flip-Flops D-Type e, in funzione del modo con cui i dati sono immessi e sono prelevati, è divisa in quattro gruppi; nella puntata precedente ci siamo occupati della modalità *SISO* (Serial In/Serial Out), caratterizzata dalla sua intrinseca compattezza, tale da richiedere solo due piedini, uno per immettere i bit d'ingresso e uno per estrarli dall'ultimo Flip-Flop della batteria, sotto il controllo del Clock.

REGISTRI A SCORRIMENTO SIPO

La modalità *SIPO* (Serial In/Parallel Out) è la logica evoluzione di quella *SISO*, ad essa strutturalmente simile ma con in più la possibilità di accedere alle singole uscite interne di ciascun Flip-Flop, disponibili su altrettanti piedini; la **figura 1** mostra la struttura di un generico SIPO Shift Register a 4 bit, dotata di ingresso seriale D (*Serial Input*, nel quale viene fatta entrare l'informazione binaria), delle uscite dei 4 D-Type coinvolti (Q_1, Q_2, Q_3 e Q_4, *Parallel Output*) e della linea opzionale di Clear, attiva bassa, che consente di predisporre a zero le medesime 4 uscite parallele. I bit in ingresso sono fatti scorrere (uno dopo l'altro da sinistra verso destra sotto l'azione del fronte di salita del Clock) attraverso ciascuna memoria elementare interna, obbligata a trasferire in uscita il valore corrente presente sul suo ingresso; sebbene sulle 4 uscite sia sempre leggibile un *nibble* (cioè una parola di 4 bit) è ragionevole supporre che l'informazione parallela possa avere una valenza significativa solo dopo un numero di periodi di clock pari a quello dei Flip-Flops utilizzati nel Registro (4, nel nostro caso), quando il primo bit immesso è arrivato in uscita all'ultima memoria elementare; e, comunque, nell'ipotesi che,della quaterna di bit introdotti in serie entri per primo quello meno significativo e per ultimo quello più significativo.

Con queste premesse il dispositivo *SIPO* può essere considerato un eccellente convertitore serie-parallelo, in grado cioè di garantire la *parallelizzazione* del dato seriale fornito in ingresso; si tratta di un'applicazione molto importante, indispensabile per gestire l'informazione binaria tra due punti molto distanti tra loro, per

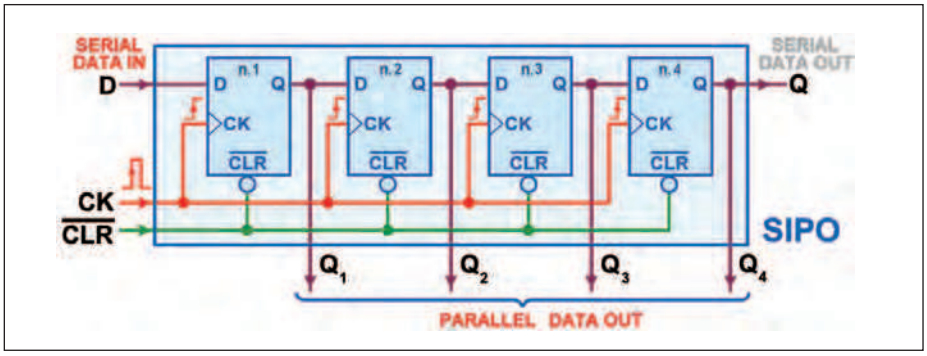
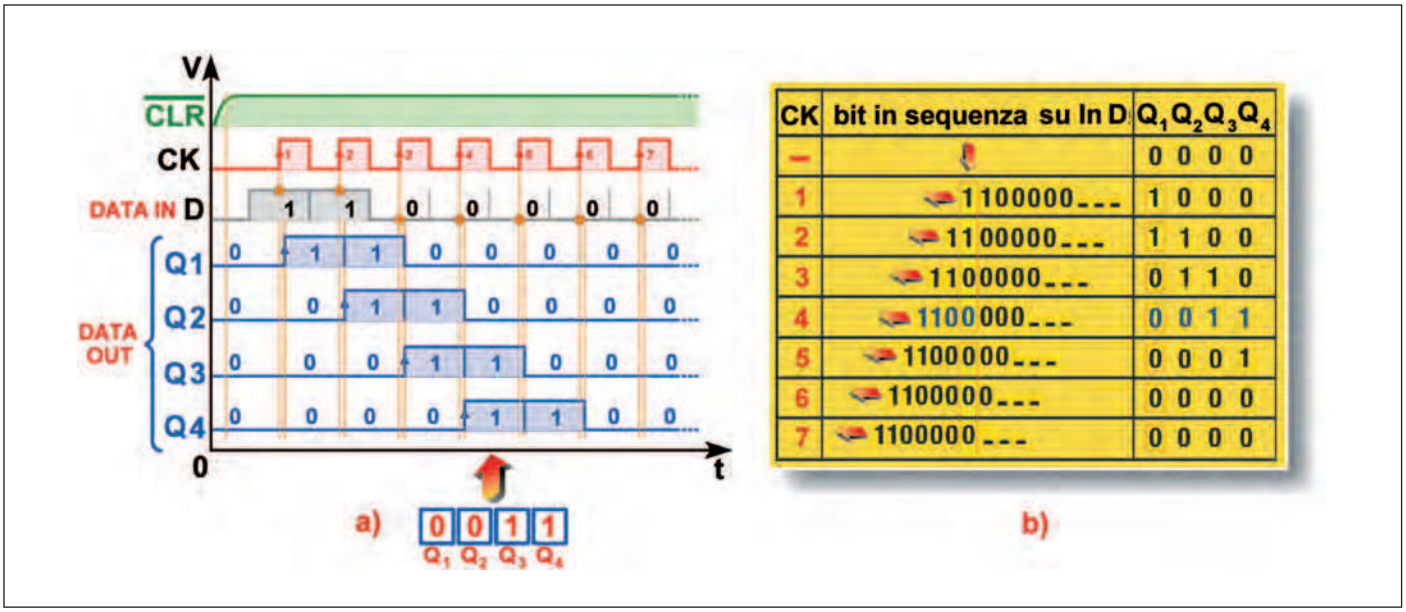


Figura 1: dettaglio funzionale di un Registro a Scorrimento in modalità SIPO.



CK	bit in sequenza su In D	$Q_1 Q_2 Q_3 Q_4$
–		0 0 0 0
1	1100000___	1 0 0 0
2	1100000___	1 1 0 0
3	1100000___	0 1 1 0
4	1100000___	0 0 1 1
5	1100000___	0 0 0 1
6	1100000___	0 0 0 0
7	1100000___	0 0 0 0

Figura 2: Diagramma Temporale nel funzionamento di un Registro a Scorrimento in modalità SIPO.

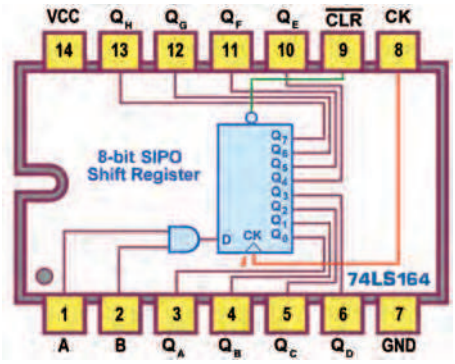


Figura 3: 8-Bit Shift Register SIPO, 74LS164: pin-out.

esempio nell'àmbito della telefonia digitale. Senza entrare troppo nel merito è facile capire che i computer (e le loro CPU) sono specializzati nell'elaborare informazioni parallele costituite da parole di dimensione multipla di 8 bit; lo stesso telefono, con l'aiuto di un convertitore analogico-digitale a 8 bit, trasforma il segnale vocale catturato dal suo microfono in una sequenza di 8000 byte al secondo: poiché la banda nominale dei segnali vocali è di 4 kHz (partendo dal fatto che la loro frequenza varia da 300 Hz a 3400 Hz) per digitalizzalo è necessario sottoporlo a campionamento con frequenza almeno doppia (appunto 8 kHz) e, affidando un byte ad ogni campione, i conti tornano. Per trasferire uno (o più) byte tra due punti distanti tra loro non è pensabile l'utilizzo di 8 fili paralleli: meglio *serializzare* la parola su un unico conduttore per poi ricostruirla nel punto d'arrivo; la conversione parallelo-serie è affidata ad un Registro a Scorrimento *PISO* (dal funzionamento opposto a quelli che stiamo trattando in questa puntata) e la trasmissione della voce così digitalizzata viene veicolata (attraverso la rete telefonica) su

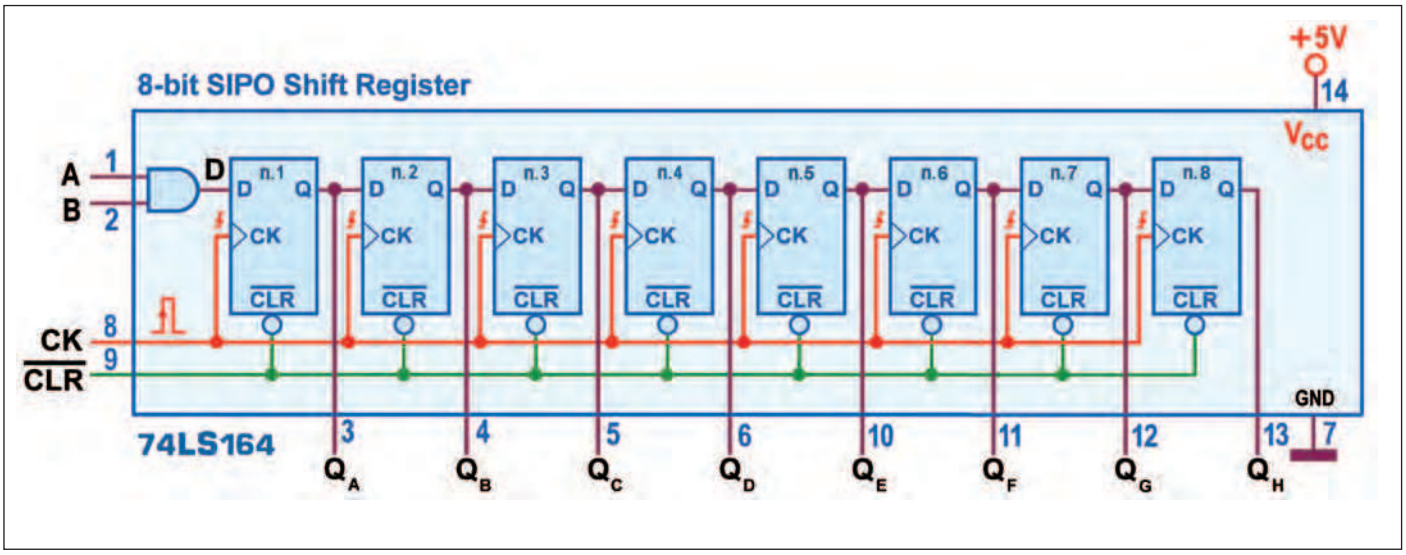
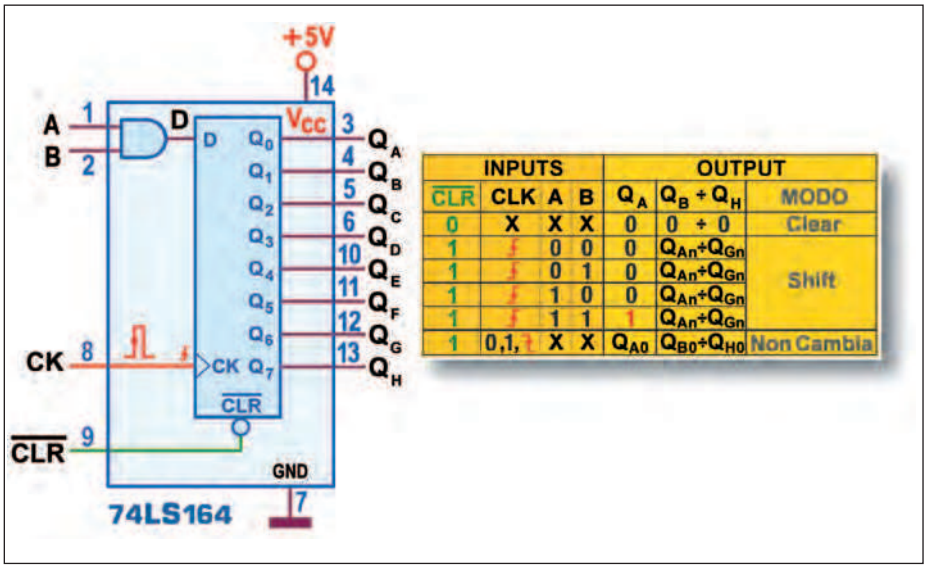


Figura 4: 8-Bit Shift Register SIPO, 74LS164: schema funzionale.

un canale seriale a 64 Kbps (8000 bytes/sec = 8000*8 bit/sec = 64K bit/sec), insieme a molti altri simili, "multiplexati" su un unico filo.
In arrivo entra in campo il nostro *SIPO*, che riporta in forma parallela l'informazione binaria seriale in arrivo; e i bytes resi disponibili dal *SIPO* sono infine ritrasformati in voce da convertitore digitale-analogico, per essere applicata al ricevitore della cornetta.
Ma torniamo al modello di **figura 1**: se è vero che dopo un numero di impulsi di clock pari al numero di stadi la sequenza di bit viene ricostruita "in blocco" sulle uscite parallele è anche vero che il primo bit di dati entrato da D (*SI*) è disponibile sull'uscita Q_4 dell'ultimo stadio che, per questo, può essere ritenuta *SO* (*Serial Outputs*) e utilizzata (se necessario) per mettere in cascata un dispositivo dello stesso tipo; per questo gli Shift Register *SIPO* possono tranquillamente essere ritenuti *SI-PO/SO*.
La **figura 2a** mostra il diagramma temporale di un *SIPO* a cui facciamo pervenire in sequenza i 4 bit 1, 1, 0, 0, seguiti da numerosi bit a 0; possiamo notare che, all'accensione (tempo 0) il segnale di Clear è mantenuto brevemente a massa, per forzare a zero tutte le uscite Parallele; non appena Clear torna a 1 il Registro è in condizioni di accettare la sequenza di fronti attivi di Clock, in corrispondenza di ciascuno dei quali l'informazione corrente presente sull'ingresso D viene trasferita dentro il Registro e fatta scorrere sulle uscite, un bit dopo l'altro.
Da notare che, per garantire il buon esito della gestione del flusso seriale, i bit in ingresso devono essere stabili per almeno un tempo pari a t_S (Setup Time) prima dell'arrivo del fronte attivo, e, subito dopo, mantenuti inalterati almeno per un tempo t_H (Hold Time); inoltre, se si desidera provvedere a una parallelizzazione dei bit in arrivo, essi devono essere proposti con una durata pari al tempo di un periodo di Clock e (possibilmente) essere campionati, dal suo fronte di salita, nel loro punto centrale. In altre parole, la frequenza di Clock deve essere pari alla velocità della sorgente seriale dei dati. L'uscita di ciascun stadio si assesterà al valore di volta in volta presente sul suo ingresso dopo un breve ritardo di propagazione, tipico del D-type utilizzato. La **figura 2b** riprende "in orizzontale" il valore corrente delle quattro uscite parallele, evidenziando il valore binario leggibile su di esse dopo l'azione di ogni fronte di Clock coinvolto nell'esperimento: il bit di volta in volta presente, nel tempo, sull'ingresso D è evidenziato in giallo al centro della colonna centrale della tabella ed è ovviamente lo stesso presente sull'uscita Q_1 del primo Flip-Flop subito dopo l'azione del fronte attivo del Clock. Naturalmente il dato parallelo ricostruito dovrà essere utilizzato (o memorizzato altrove) prima dell'arrivo del quinto fronte attivo di Clock, altrimenti il suo valore originale sarà perduto e sostituito con altrettanti nuovi bit, assunti dall'esterno; questa nefasta opportunità è definita "framing error", ma il tempo a disposizione per evitarla è comunque notevole (un periodo di clock, tra il quarto e il quinto fronte di salita).



INPUTS				OUTPUT		
$\overline{CLR}$	CLK	A	B	Q_A	$Q_B \div Q_H$	MODO
0	X	X	X	0	0 ÷ 0	Clear
1	↑	0	0	0	$Q_{An} \div Q_{Gn}$	Shift
1	↑	0	1	0	$Q_{An} \div Q_{Gn}$	
1	↑	1	0	0	$Q_{An} \div Q_{Gn}$	
1	↑	1	1	1	$Q_{An} \div Q_{Gn}$	
1	0,1,↓	X	X	Q_{A0}	$Q_{B0} \div Q_{H0}$	Non Cambia

Figura 5: 8-Bit Shift Register SIPO, 74LS164: schema pratico.

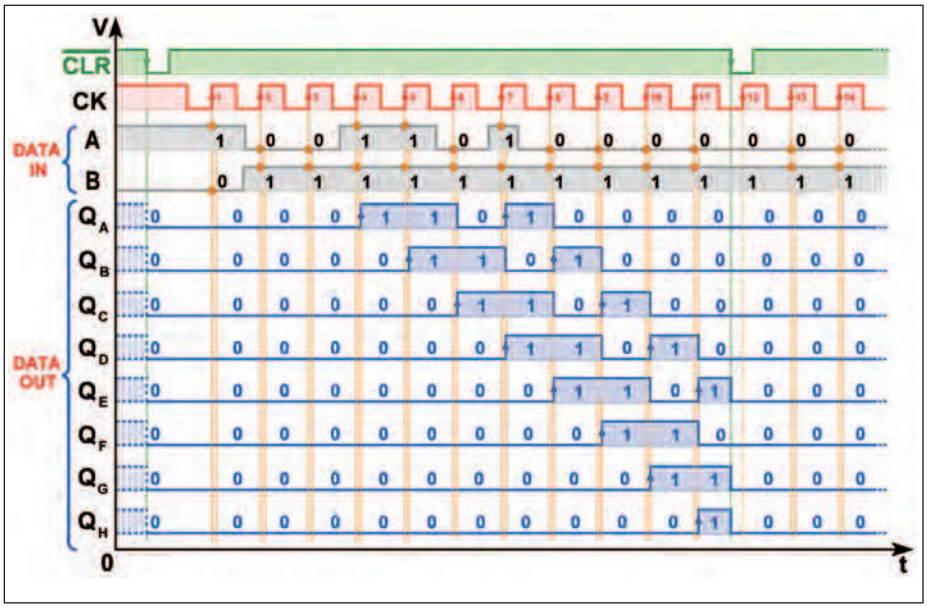


Figura 6: 8-Bit Shift Register SIPO, 74LS164: diagramma temporale.

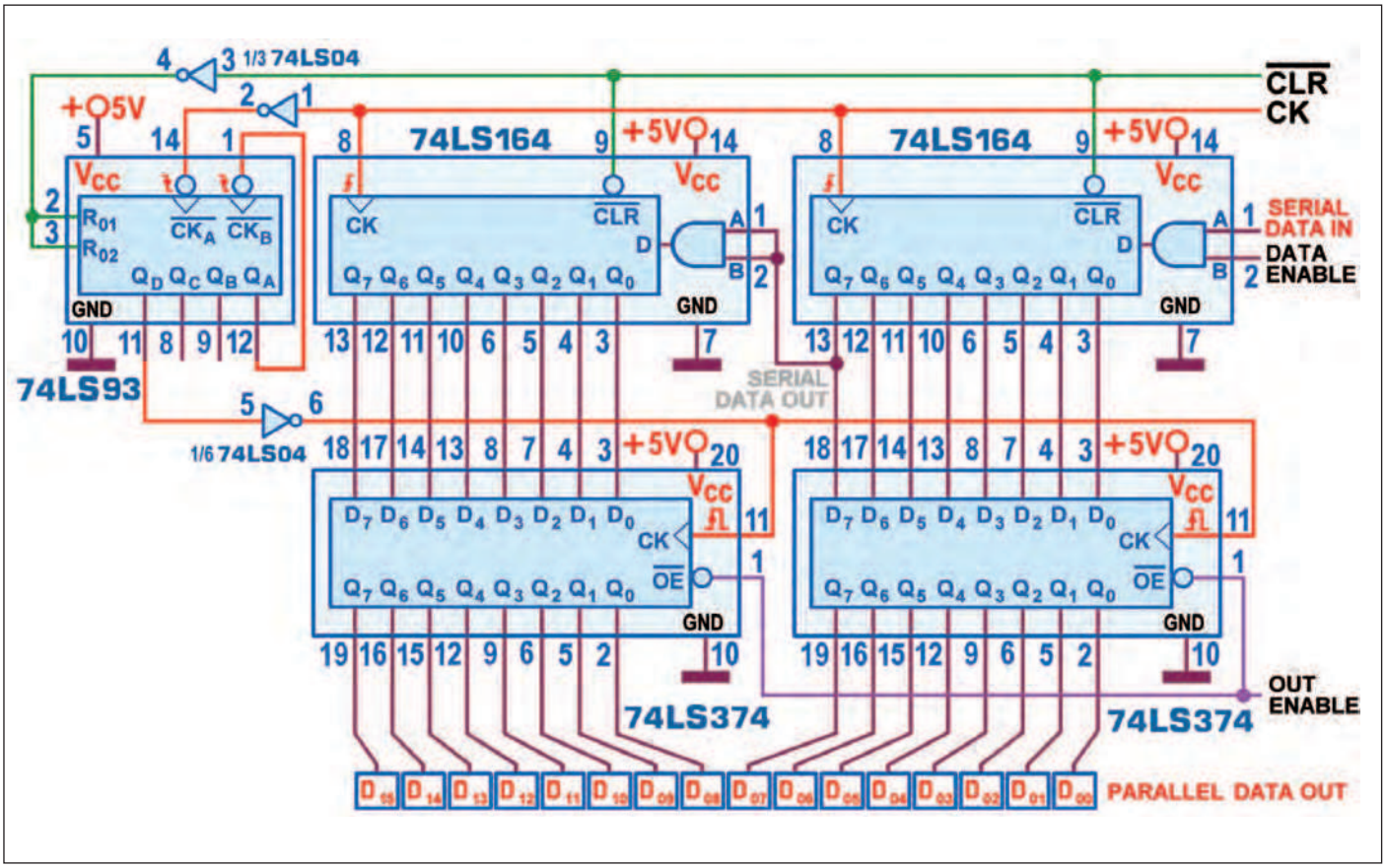


Figura 7: 8-Bit Shift Register SIPO, 74LS164: versione a 16 bit con memoria opzionale.

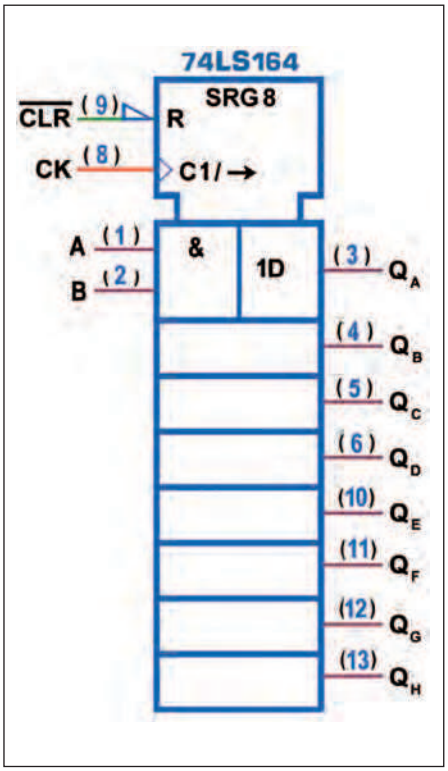


Figura 8: 8-Bit Shift Register SIPO, 74LS164: simbolo logico ANSI/IEEE Std. 91-1984.

REGISTRI A SCORRIMENTO SIPO (TTL): 74LS164, 74LS594, 74LS595, 74LS596, 74LS599, 74LS673

Il 74LS164 (un 8-Bit SIPO Shift Register) è un Registro a Scorrimento ad alta velocità (il più utilizzato della categoria *SIPO*), con ingresso seriale dei dati e un'uscita per ognuno dei suoi 8 stadi; la **figura 3** mostra il suo pinout. La **figura 4** offre lo schema funzionale che aiuta a comprendere i dettagli della sua struttura; si nota che tutti gli elementi di memoria hanno in comune la linea di Clock (attiva sul fronte salita) e la linea di Clear, come al solito asincrona e del tutto prioritaria. Dispone di due ingressi seriali, A e B, coordinati da una AND, per stabilire l'effettivo valore del *Serial Input*, trasferito sull'uscita del primo Flip-Flop D-Type all'arrivo del primo fronte di salita del Clock; se su A o su B (o su entrambi) viene forzato uno 0, il valore predisposto sul *SI* interno D è uno 0, mentre affinchè sia un 1 è necessario che entrambi gli ingressi A e B siano a 1; da un altro punto di vista si può dire che uno di questi ingressi può essere utilizzato come abilitazione (attiva alta) per permettere l'immissione dei dati attraverso l'altro ingresso: la logica di gating (assicurata dalla AND) fa sì che il valore di A passi inalterato su D quando B=1 (porta aperta) e lasci D=0 quando B=0 (porta chiusa).

Ovviamente tutto funzionerà perfettamente anche a ruoli invertiti, postando i dati su B e affidando ad A il compito di abilitarne o inibirne l'ingresso; di fatto uno dei due ingressi potrebbe anche non essere utilizzato e lasciato scollegato: sebbene (avendo a che fare con una logica TTL) esso "senta" un 1 d'ufficio (e tenga aperta la porta AND) è sempre fortemente consigliato evitare situazioni d'instabilità collegandolo alla tensione di alimentazione positiva oppure collegando assieme entrambi gli ingressi.

La figura 5 propone lo schema pratico con la Tabella di Verità, che sintetizza quanto detto: l'azione prioritaria di Clear (cioè con CLR=0) azzera tutte le uscite, indipendentemente dal valore assunto

dalle altre linee; quando Clear non è attivo (CLR=1), in presenza del fronte di salita del Clock l'uscita Q_A assume il valore 0 se almeno un ingresso A o B è a 0, oppure il valore 1 se entrambi gli ingressi sono a 1; le uscite dei rimanenti Flip-Flops, da Q_B a Q_H, assumono il valore che avevano le uscite dello stadio precedente (rispettivamente indicate con la notazione Q_{An}, ... , Q_{Gn} a indicare lo spostamento a destra di un bit) prima dell'arrivo della recente transizione di Clock.

L'ultima riga della Tabella ribadisce lo stato d'attesa del componente, durante il quale tutte le uscite, da Q_A a Q_H, mantengono il valore assunto dopo l'arrivo dell'ultima transizione di Clock o per effetto di un precedente azzeramento, indicato rispettivamente con la notazione da Q_{A0} a Q_{H0}.

La disponibilità dei diagrammi temporali di un dispositivo sequenziale (come il nostro *SIPO*) è sempre una fortuna: niente come questo strumento è in grado di spiegare il suo funzionamento; in **figura 6** è mostrato quello offerto dai datasheet per il 74LS164.

Ecco alcune considerazioni: a) all'accensione (tempo 0) il valore delle uscite non è prevedibile; per questo è sempre consigliabile garantire una breve azione di Clear (di solito assicurata con una rete di reset automatico), per partire con tutte le uscite a 0; b) mentre Clear è inattivo ogni fronte di salita del Clock provvede a interpretare il valore corrente del prodotto logico (AND) degli ingressi A e B, facendolo scorrere all'interno attraverso tutti gli 8 stadi del Registro; da notare che in questo esempio i valori proposti su entrambe le linee di dato sono del tutto asincroni rispetto al Clock cosicché, se il loro prodotto fosse a 0 e passasse stabilmente a 1 nel medesimo istante in cui viene prodotto il fronte attivo, è evidente che in uscita al primo stadio verrebbe trasferito uno 0 e solo all'arrivo del successivo fronte essa passerebbe a 1; **c**) un eventuale nuovo intervento del segnale di Clear riporta tutte le uscite a 0 e annulla tutti i fronti attivi forniti mentre è a livello basso.

La **figura 7** mostra come realizzare un Registro a scorrimento a 16 bit mettendo in cascata due 74LS164, cioè collegando l'uscita Q_H=Q_7 dell'ultimo stadio del primo dispositivo con gli ingressi (uniti tra loro) del secondo; il flusso seriale d'ingresso è affidato alla linea A, abilitato a entrare se B=1. Se si suppone che i bit in ingresso siano sincronizzati con il Clock (cioè proposti con una durata pari al suo periodo e *campionati* possibilmente nel loro punto centrale), dopo 16 fronti di salita sulle uscite sono leggibili i 2 byte frutto della conversione serie-parallelo, a disposizione per una eventuale elaborazione per un intero periodo di Clock, cioè fino all'arrivo del diciassettesimo impulso attivo, dopodiché l'informazione sarà distrutta dai nuovi bit in arrivo, per tornare significativa solo dopo ulteriori 16 transizioni.

Per evitare questo fastidioso evento, il progetto è stato arricchito con ulteriori integrati: la presenza di due 74LS374 permette di memorizzare lo stato dei due shift register solo quando sulle loro uscite è stata ricostruita la parola finale a 16 bit; il processo deve cominciare con un breve livello basso sulla linea di Clear, che provvede ad azzerare le 16 uscite del registro e le 4 uscite di un contatore bi-

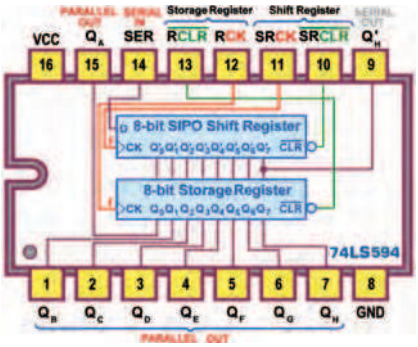


Figura 9: 8-Bit Shift Register SIPO with Output Latches, 74LS594: pin-out.

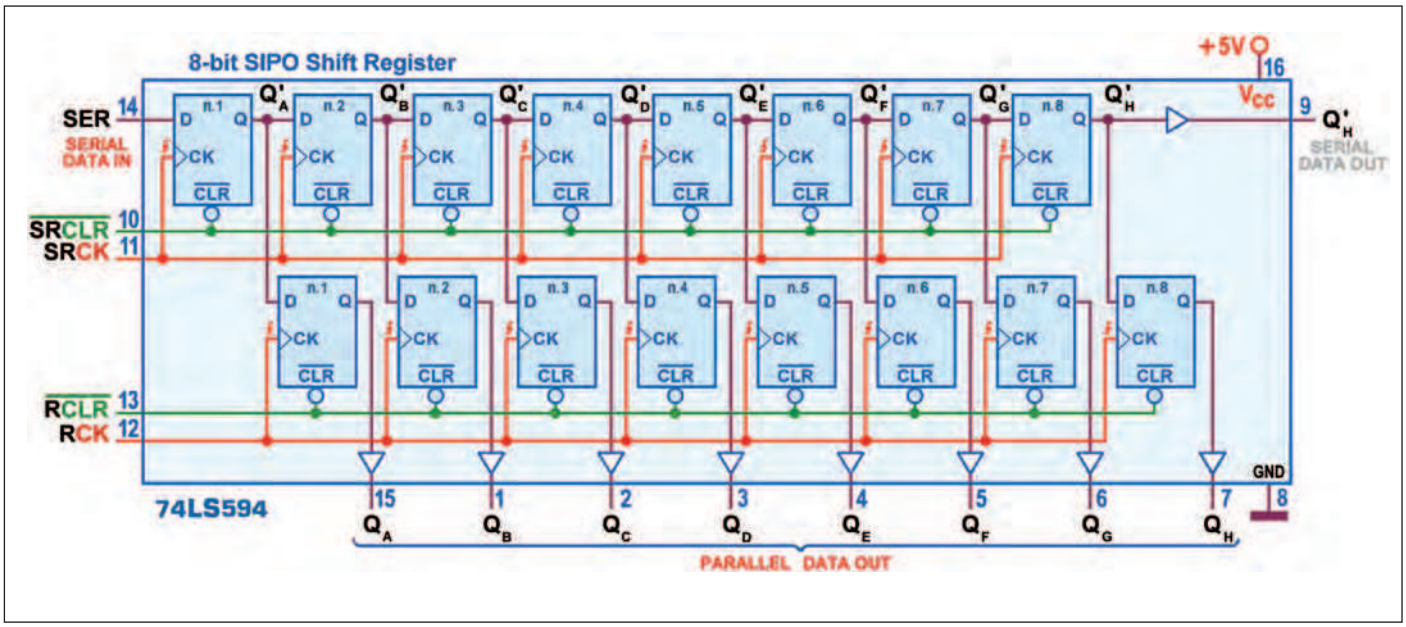


Figura 10: 8-Bit Shift Register SIPO with Output Latches, 74LS594: schema funzionale.

nario (un 74LS93); non appena questa linea torna alta (rimanendovi per sempre), dopo ogni fronte attivo i bit cominciano a entrare (supposto B=1) e il contatore incrementa il suo valore a 4 bit. All'arrivo del sedicesimo impulso la parola desiderata è stata ricostruita sulle 16 uscite e il contatore passa da 1111 a 0000, riazzerandosi; in particolare l'uscita Q_D (sul suo pin 11) passa da 1 a 0 e, in virtù della presenza dell'inverter, fornisce al Clock delle memorie il fronte di salita necessario per bloccare l'informazione attualmente presente sullo shift, mantenendola a disposizione fino a quando un ciclo completo di altri 16 impulsi riproporrà le stesse condizioni, appena descritte.

La **figura 8** propone infine il simbolo logico previsto dallo standard IEEE: a) la dicitura SRG8 è la qualifica generale per un Registro a Scorrimento a 8 bit; b) l'ingresso di Clock (nella parte alta del simbolo, destinata alla sezione di controllo) ha due funzioni interne: il numero 1 dell'etichetta C1 sta a indicare che esso esercita il controllo di qualsiasi cosa (nella parte bassa del simbolo) con prefisso 1, cioè identifica l'area dati 1D (l'etichetta dei dati in ingresso al primo stadio); la seconda funzione, il segno "/" seguito da una "freccia a destra", indica lo spostamento da sinistra a destra dei dati all'interno del registro di scorrimento; c) al primo degli 8 stadi che compongono la parte inferiore del simbolo è affidato il dettaglio (da condividere implicitamente con gli altri) del controllo esercitato sui due ingressi di dato A e B, concorrenti nel rettangolo con il simbolo "&" (identificatore della logica AND che li coordina); d) ciascuno degli otto stadi evidenzia sul lato destro le rispettive uscite.

La potenza massima dissipata dal 74LS164 è di 135 mW; la corrente massima I_{OL} assorbita in uscita è nella norma (8 mA, e 16 mA per la standard); i ritardi di propagazione massimi (con carico di 15pF/2000ohm) sono t_{PLH}=27 ns e t_{PHL}=32 ns ; il Setup Time minimo vale t_S=15 ns e l'Hold Time minimo vale t_H=5 ns. La frequenza massima di lavoro è di 25 MHz. La serie TTL prevede anche 4 componenti (simili tra loro) pensati per soddisfare le stesse esigenze garantite dal progetto illustrato in precedenza:

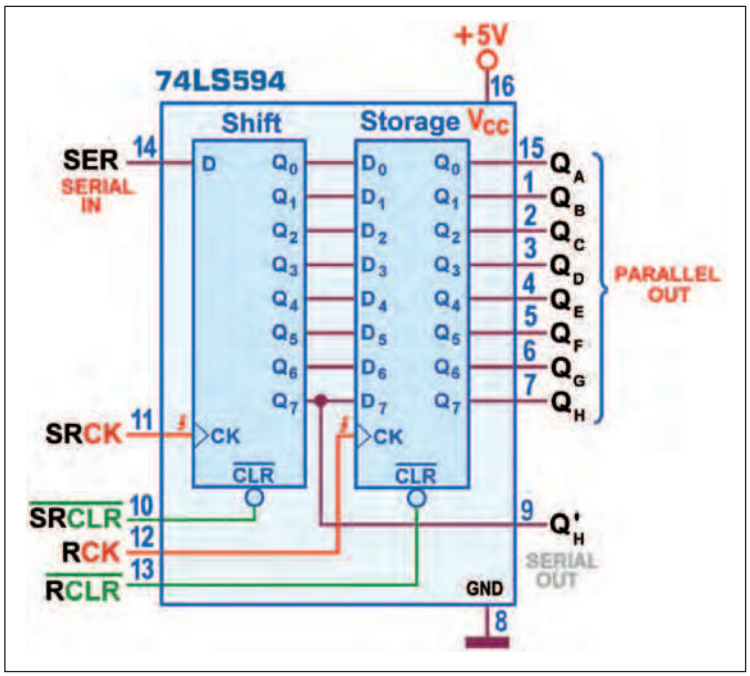


Figura 11: 8-Bit Shift Register SIPO with Output Latches, 74LS594: schema pratico.

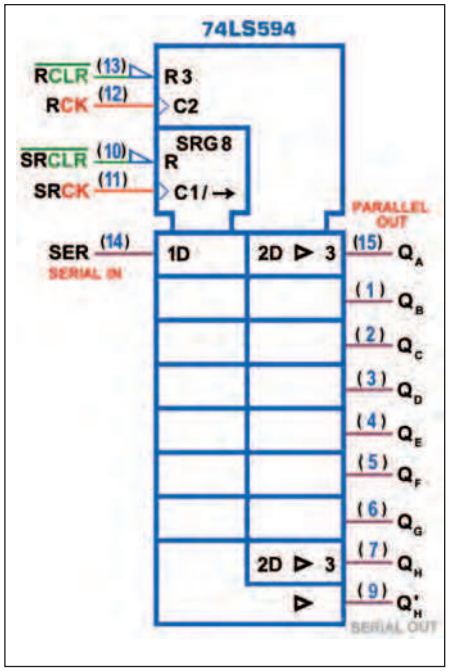


Figura 12: 8-Bit Shift Register SIPO with Output Latches, 74LS594: simbolo logico ANSI/IEEE Std. 91-1984.

contengono infatti un Registro a Scorrimento (*Shift Register*) a 8 bit di tipo *SIPO* con le uscite collegate internamente agli ingressi di un Registro Parallelo (*Storage Register*) della stessa capacità: le due unità di memoria hanno linee di Clock separate così che i dati possano essere spostati all'interno della sezione superiore (su ogni fronte di salita della linea SRCK, *Shift Register Clock*) senza alterare l'informazione presente sulle linee d'uscita (*Parallel Data Output*) della sezione inferiore; non appena lo Shift renderà disponibile un nuovo dato a 8 bit (cosa possibile solo dopo l'arrivo dell'ottavo impulso di SRCK) dovrà essere prodotto un fronte di salita sulla linea RCK (*Storage Register Clock*) per poterlo memorizzare al posto del precedente, mantenendolo costante fino a quando il ciclo descritto non verrà ripetuto.

Per sincronizzare gli eventi che governano questa procedura il Registro *SIPO* è dotato di linea di Reset autonoma, linea SRCLR (*Shift Register Clear*), attiva bassa; nel normale funzionamento questa linea è tenuta costantemente alta, forzandola brevemente a 0 (di solito qualche millisecondo) solo all'accensione del circuito, per predisporre a 0 tutte le sue uscite prima di autorizzare gli 8 impulsi di SRCK, chiamati a "*parallelizzare*" gli 8 bit immessi dall'ingressi SER (*Serial Input*); sulla medesima memoria è disponibile esternamente (sul pin9) anche l'uscita del suo ultimo stadio, Q_H'=Q_7' , utilizzabile come uscita seriale (*Serial Output*), ideale per favorire il collegamento in cascata un eventuale altro dispositivo.

Il primo dei quattro componenti è il 74LS594 (un 8-Bit SIPO Shift Register with Output Latches) ed è caratterizzato dal fatto di avere le uscite esterne bufferizzate ma normali, cioè interessate dai classici due stati; la **figura 9** mo-

stra il suo pinout. La **figura 10** offre lo schema funzionale dal quale risulta evidente la citata presenza di linee separate di Clock (entrambe attive sul fronte di salita) e di Clear (entrambe attive basse) sia per il Registro a Scorrimento (*Shift*) sia per quello di Memorizzazione (*Storage*); questa disponibilità rende molto versatile la gestione del componente.
La **figura 11** propone lo schema pratico,

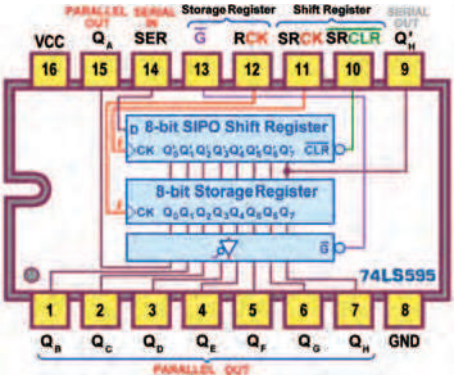


Figura 13: 8-Bit Shift Register SIPO with Output Latches, 74LS595: pin-out.

compatto e molto utile nella stesura dei nostri progetti. In **figura 12** vediamo infine il simbolo logico previsto dallo standard IEEE; in questo caso il simbolo è diviso in due sezioni, sia nella parte alta (di controllo) sia in quella bassa (controllata): l'etichetta C1 seguita dal segno "/" e da una "freccia a destra" sta ad indicare che il Clock (ad essa associato) influenza i segnali della parte bassa a sinistra (caratterizzati dal prefisso 1), cioè i dati 1D in ingresso allo *Shift Register* a 8 bit, come sottolinea la dicitura SRG8, provocandone lo spostamento da sinistra a destra; l'etichetta C2 indica che il Clock agisce sui segnali (con prefisso 2) della parte bassa a destra, cioè sui dati 2D in ingresso allo *Storage Register*, per i quali non è previsto (ovviamente) alcuno scorrimento verso destra ma le cui uscite sono bufferizzate (come testimonia il triangolo con punta a destra. Su di esse (marcate dell'etichetta 3) agisce coerentemente la linea di Clear R3.
La potenza massima dissipata dal 74LS594 è di 325 mW; la corrente massima I_{OL} assorbita in uscita è di 24 mA (per le 8 Parallel Outputs) e 16 mA (per la Serial Output Q_H'); i ritardi di propagazione massimi previsti tra SRCK e uscita seriale (con carico di 15pF/2000ohm) sono di 18 ns (t_{PLH}) e 23 ns (t_{PHL}) e quelli tra RCK e uscite parallele (con carico di 45pF/667ohm) sono di 18 ns (t_{PLH}) e 30 ns (t_{PHL}); leggermente più alti i tempi necessari per azzerare le uscite (rispettivamente con SRCLR e RCLR); il valore minimo di Setup Time vale t_S=20 ns, mentre quello di Hold Time è trascurabile. La frequenza massima di scorrimento (SRCK) è di 20 MHz, mentre quella di memorizzazione (RCK) è di 25 MHz.
Del componente appena descritto esiste una versione funzionalmente identica (il 74LS599, ancora uno Shift Register SIPO a 8 bit collegato internamente a uno Storage Register) ma con uscite parallele di tipo open collector; per esso valgono le considerazioni fatte per il 74LS594, nonché gli schemi di pinout **(figura 9)**, funzionale **(figura 10)**, pratico **(figura 11)**.
Anche il simbolo logico IEEE è sostanzialmente uguale a quello di **figura 12**, ma contiene un dettaglio differente: nella sezione in basso a destra, relativa agli 8 blocchi che rappresentano gli 8 D-Type dello *Storage Register*, sull'estrema destra apparirà ora un piccolo rombo, destinato a giustificare le uscite open collector, al posto del "triangolo rivolto verso destra", tipico delle uscite bufferizzate.
La potenza massima dissipata dal 74LS599 è di 275 mW, la corrente massima I_{OL} assorbita in uscita è la stessa del 74LS594, mentre i ritardi di propagazione massimi sono leggermente più

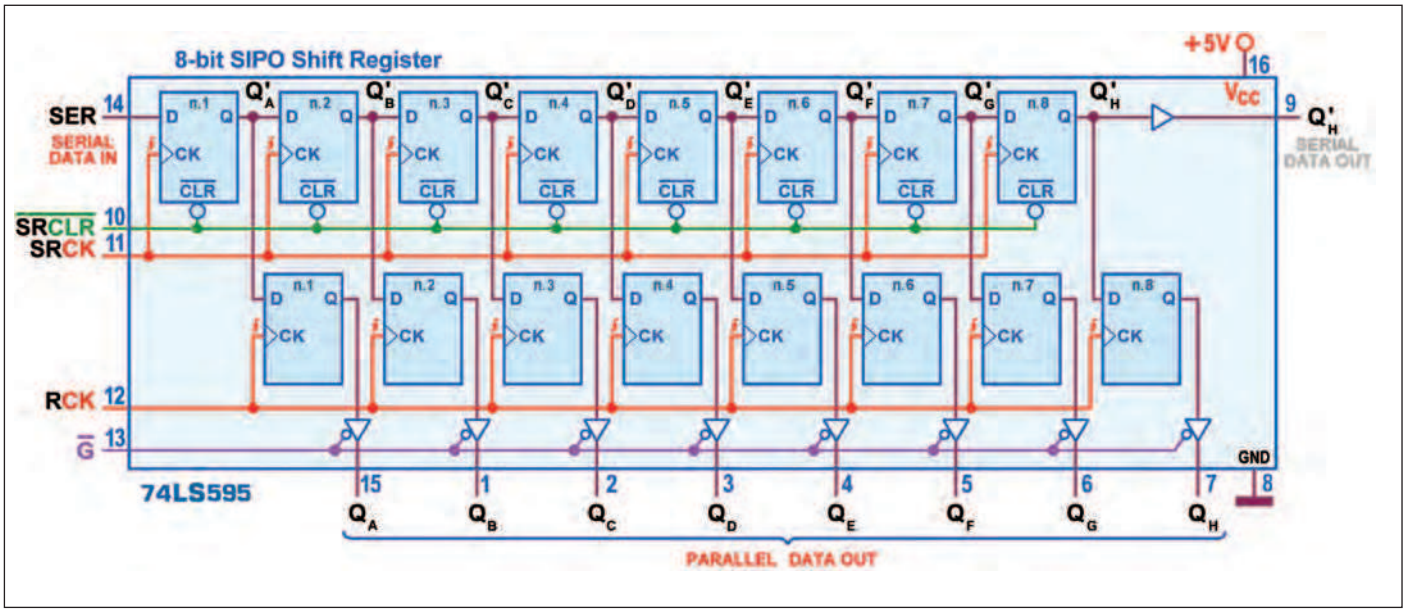


Figura 14: 8-Bit Shift Register SIPO with Output Latches, 74LS595: schema funzionale.

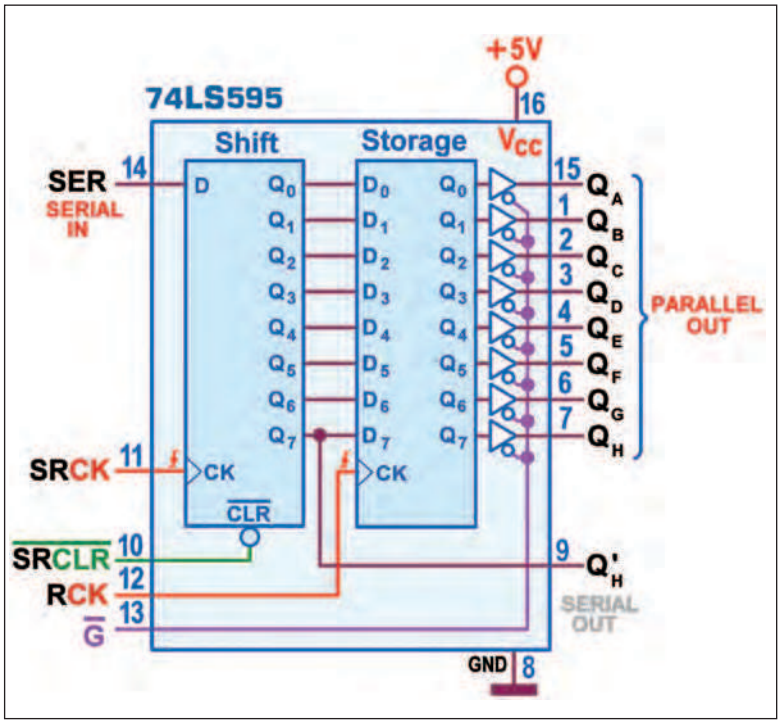


Figura 15: 8-Bit Shift Register SIPO with Output Latches, 74LS595: schema pratico.

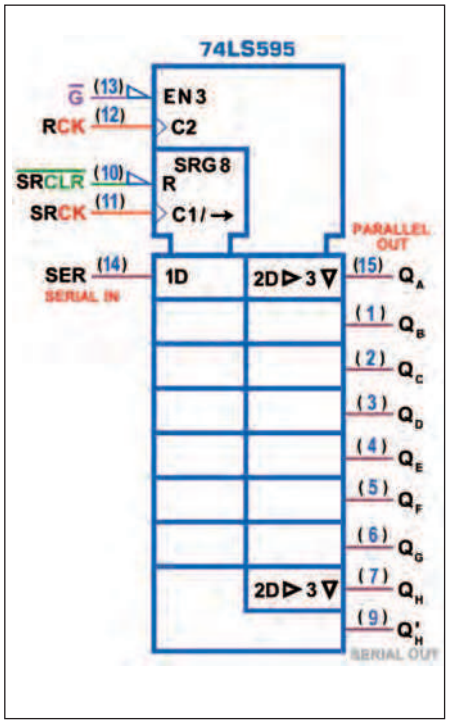


Figura 16: 8-Bit Shift Register SIPO with Output Latches, 74LS595: simbolo logico ANSI/IEEE Std. 91-1984.

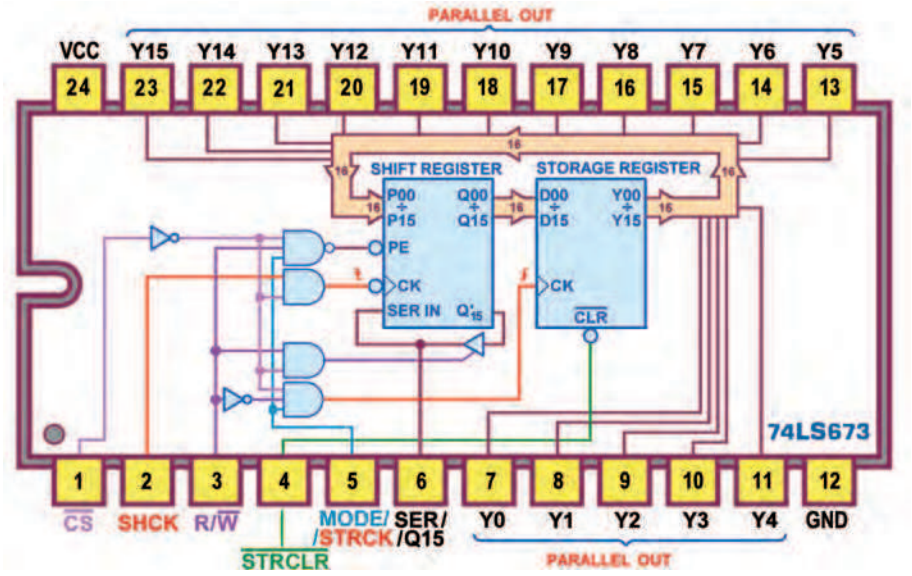


Figura 17: 16-Bit Shift Register SIPO with 16-Bit PO Storage Register, 74LS673: pin-out.

alti; stessi valori invece per Setup Time minimo (Hold Time trascurabile) e frequenze massime di lavoro.

Il 74LS595 è sempre un 8-Bit SIPO Shift Register with Output Latches, funzionalmente identico alla coppia precedente, dalla quale differisce per la mancanza della possibilità di azzerare preventivamente le 8 uscite dello *Storage Register*, cioè della linea RCLR, sostituita dalla linea G, attiva bassa, necessaria per abilitare l'informazione sulle 8 uscite, ora di tipo 3-state, o per forzarle in alta impedenza (con G=1); rimane la possibilità di azzerare le 8 uscite interne dello *Shift Register*, essendo ancora presente la linea SRCLR, attiva bassa; la **figura 13** mostra il pinout di questo componente.

Nella **figura 14** vediamo lo schema funzionale, che conferma la presenza di linee di clock separate per le due unità di memoria, entrambe attive sul fronte di salita ; la prima (SRCK) è necessaria per spostare i dati dentro lo *Shift Register*, senza influenzare le 8 uscite dello *Storage Register*, eventualmente aggiornabili per azione di RCK, di solito dopo l'ottava transizione attiva di SRCK, con un nuovo dato a 8 bit pronto per essere memorizzato.

Oltre al necessario *Serial Input* (SER), è sempre disponibile anche una *Serial Output* (sfruttando l'uscita Q_H' del suo ultimo stadio), utile per porre in cascata un altro 74LS595.

La **figura 15** propone lo schema pratico; si noti come la presenza del pin 13 di Out Enable, G, renda particolarmente conveniente l'utilizzo di questo componente nella realizzazione del Registro a scorrimento a 16 bit descritto in **figura 7**.

In **figura 16** vediamo il simbolo logico previsto dallo standard IEEE; le differenze con quello del 74LS594 sono quelle già rimarcate: in particolare, la presenza (nella sezione di controllo) di EN3 (associato a G) invece di R3 (associato a RCLR, ora assente), comunque riferita (nella sezione controllata, in basso a destra) alle uscite (con l'etichetta 3) dello *Storage Register*, ora caratterizzate da un triangolo con punta verso il basso, per indicare che sono di tipo 3-state.

Anche del 74LS595 esiste una versione

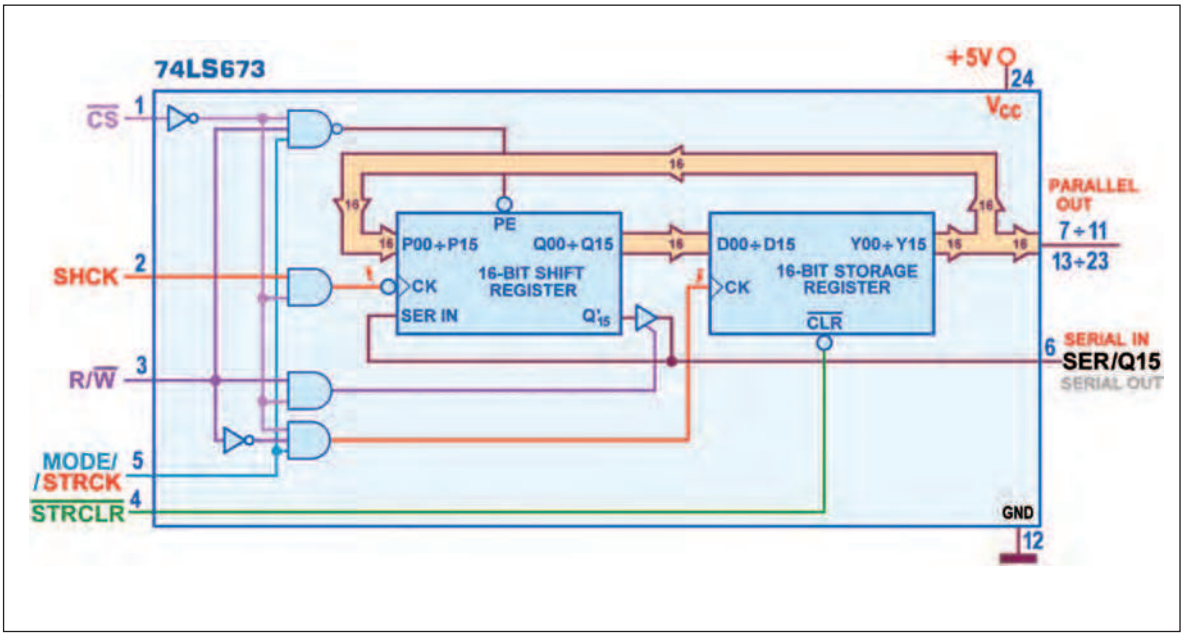


Figura 18: 16-Bit Shift Register SIPO with 16-Bit PO Storage Register, 74LS673: schema funzionale.

funzionalmente identica ma con uscite di tipo open collector; si tratta del 74LS596 per il quale valgono le stesse considerazioni e gli stessi schemi di pinout **(figura 13)**, funzionale **(figura 14)**, pratico **(figura 15)** e del simbolo IEEE, uguale a quello di **figura 16**, a eccezione del piccolo rombo al posto del "triangolo rivolto verso il basso", per qualificare correttamente la natura delle uscite parallele, ora open collector. Le caratteristiche elettriche e dinamiche di questa coppia di componenti sono sostanzialmente le stesse sottolineate per la coppia 74LS594/74LS599.

L'ultimo componente della serie TTL è il 74LS673, un 16-Bit SIPO/SO Shift Register with 16-Bit PO Storage Register; la proprietà di gestire 16 bit in unico chip richiede una sua maggiore dimensione, mostrata nel pinout di **figura 17**. Si tratta di un dispositivo molto sofisticato; come gli altri trattati finora, dispone di linee di clock separate per la sezione *Shift Register* (SHCK, attivo sul fronte di discesa) e per quella *Storage Register* (STRCK, attivo sul fronte di salita); la linea d'ingresso dati (*Serial Input*, SER) è la stessa utilizzata per l'uscita seriale (*Serial Output*, Q_{15}'); per poter gestire questa bidirezionalità è predisposta una linea R/W che abilita la *scrittura* del registro di scorrimento (se posta a 0, scollegando di fatto la linea di input/output SER/Q15, forzata in alta impedenza, e consentendo l'immissione di dati seriali)

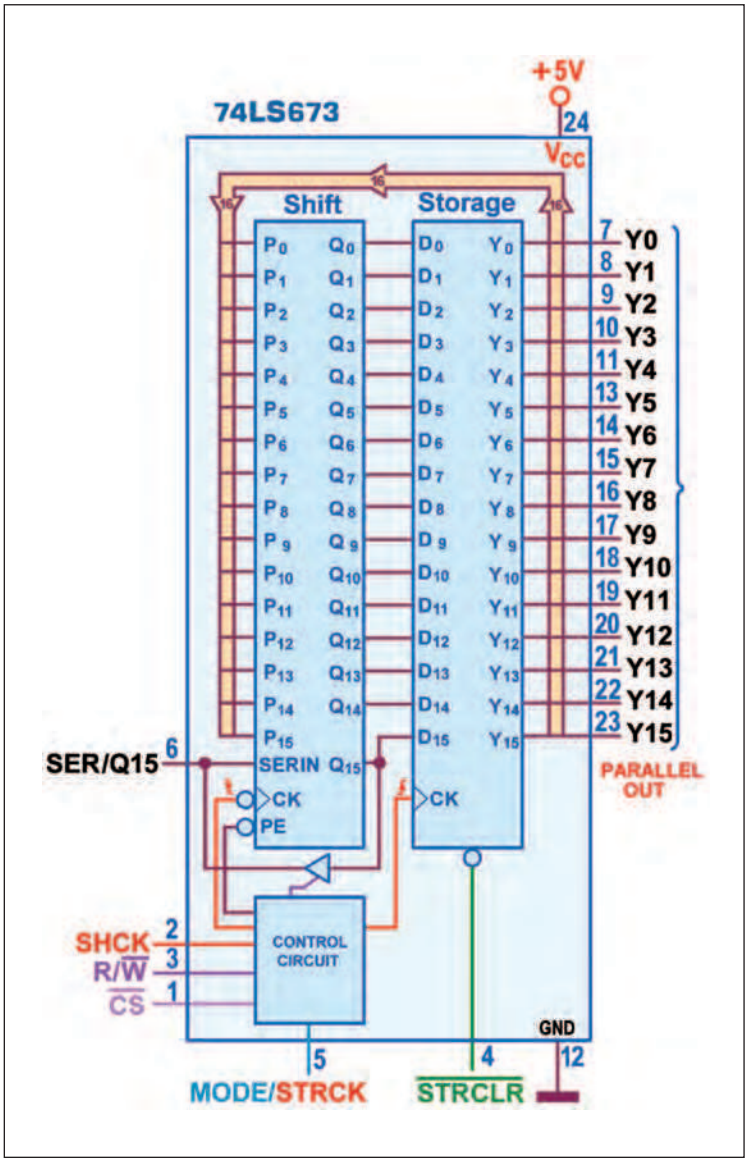


Figura 19: 16-Bit Shift Register SIPO with 16-Bit PO Storage Register, 74LS673: schema pratico.

convertitori USB-seriale

compatti, performanti e convenienti

MULTI-1U
Convertitore USB 1.1 a **1 Porta** RS232 o 422/485

MULTI-2U
Convertitore USB 1.1 a **2 Porte**
RS232 o 422/485

MULTI-8U
Convertitore USB 1.1 a **8 Porte**
RS232 o 422/485

MULTI-4U
Convertitore USB 1.1
a **4 Porte**
RS232 o 422/485

CODICE MIP 2814090

SystemBase

Per maggiori informazioni visita Il sito **www.elettroshop.com** oppure chiama il numero **02/66504794**

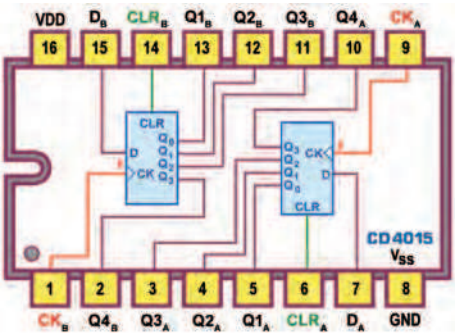


Figura 20: Dual 4-Bit SIPO Static Shift Register, 4015: pin-out.

oppure la sua lettura (se posta a 1, rendendo disponibile su SER/Q15 il dato, in uscita dal suo ultimo stadio).
Come succede nei chip di memoria o in quelli maggiormente organizzati, dispone inoltre di un CS (*Chip Select*, attivo basso) che controlla in blocco le sue funzioni vitali: un livello logico alto su di esso disabilita entrambe le linee di Clock e forza comunque la linea SER/Q15 in alta impedenza, rendendo di fatto inaccessibile il componente sia in scrittura sia in lettura; per evitare false transizioni è tuttavia necessario fare molta attenzione, mantenendo basso il Clock SHCK dello *Shift Register* durante la transizione basso-alto di CS e il Clock STRCK dello *Storage Register* durante la transizione alto-basso di CS.
Solo lo *Storage Register* è dotato di una linea di Reset autonoma (STRCLR, attiva bassa, l'unica a non essere disattivata dal *Chip Select*), chiamata ad azzerarne le uscite in modo asincrono, se forzata a 0. La **figura 18** riassume queste prerogative nello schema funzionale; possiamo notare che il Registro di Memorizzazione può ancora essere caricato in parallelo con il dato dello *Shift Register* per rendere disponibile lo stato delle sue uscite interne attraverso quelle esterne (*Parallel Output*); ma in questo caso anche le uscite dello *Storage Register* possono essere caricate in parallelo

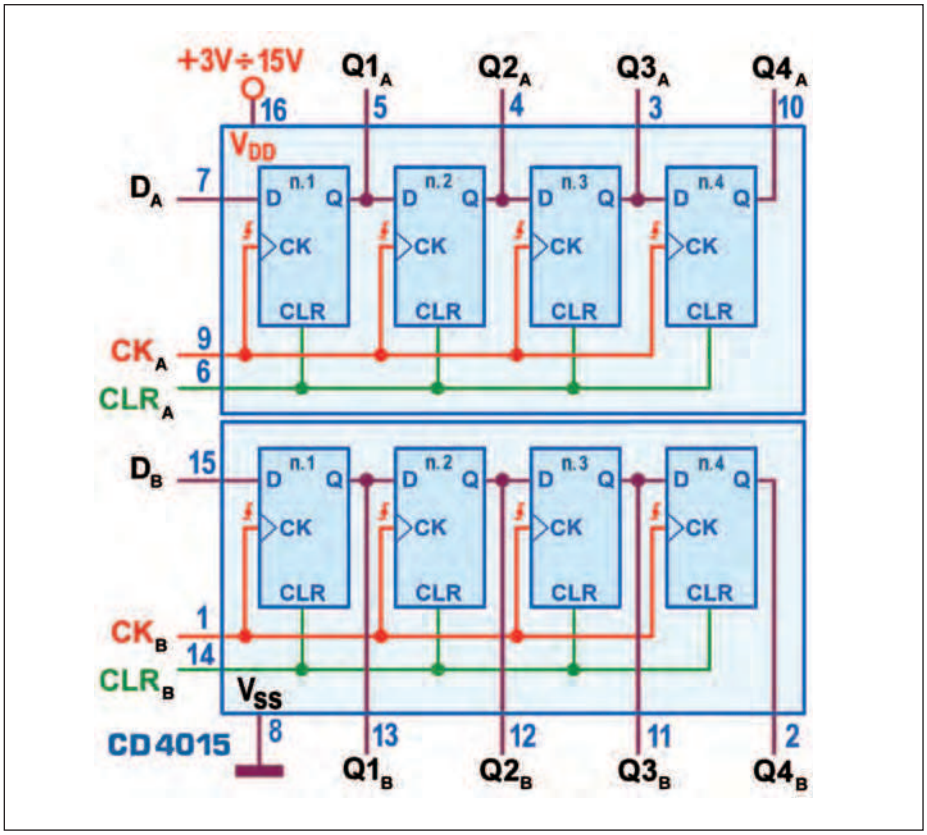
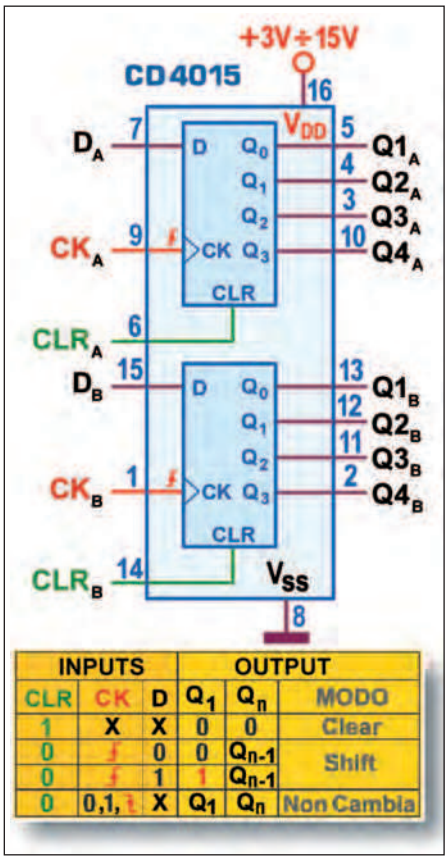


Figura 21: Dual 4-Bit SIPO Static Shift Register, 4015: schema funzionale.



INPUTS			OUTPUT		
CLR	CK	D	Q_1	Q_n	MODO
1	X	X	0	0	Clear
0	↓	0	0	Q_{n-1}	Shift
0	↓	1	1	Q_{n-1}	
0	0,1,↑	X	Q_1	Q_n	Non Cambia

Figura 22: Dual 4-Bit SIPO Static Shift Register, 4015: schema pratico.

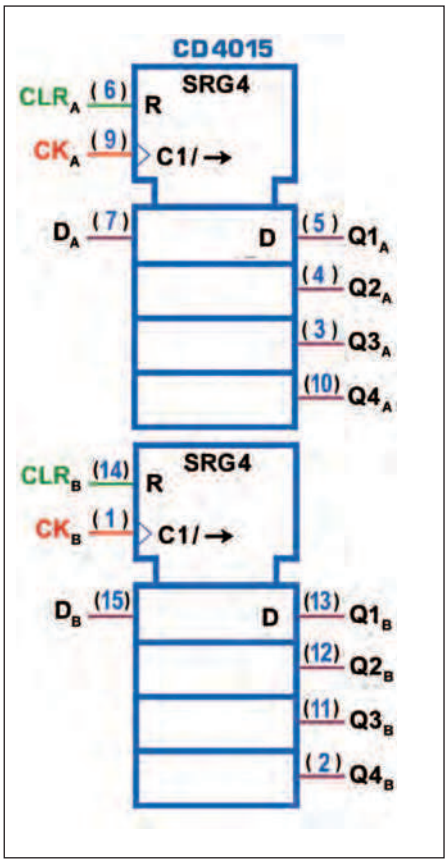


Figura 23: Dual 4-Bit SIPO Static Shift Register, 4015: simbolo logico ANSI/IEEE Std. 91-1984.

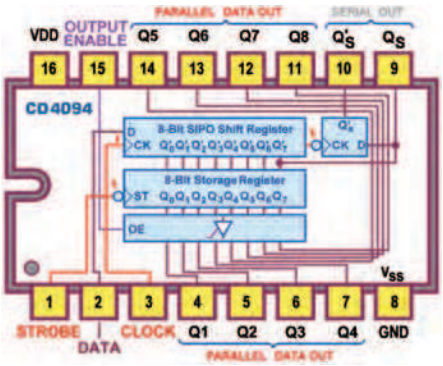


Figura 24: 8-Bit Shift Register SIPO with Output Latches, 4094: pin-out.

(*Load*) in ingresso al Registro a Scorrimento, attivando il segnale interno PE subordinato dall'azione coordinata degli altri segnali di controllo; in particolare: il segnale MODE/STRCK (a 0, in condizioni di funzionamento normale) per l'azione di caricamento (durante la quale non è consentito lo spostamento dei dati) va posto a 1 con CS =0, R/W=1 e STRCLR=1.

La **figura 19** propone lo schema pratico; il simbolo logico previsto dallo standard IEEE (piuttosto ingombrante, per la presenza dei numerosi segnali coinvolti) è in questo caso omesso per non appesantire troppo la trattazione.

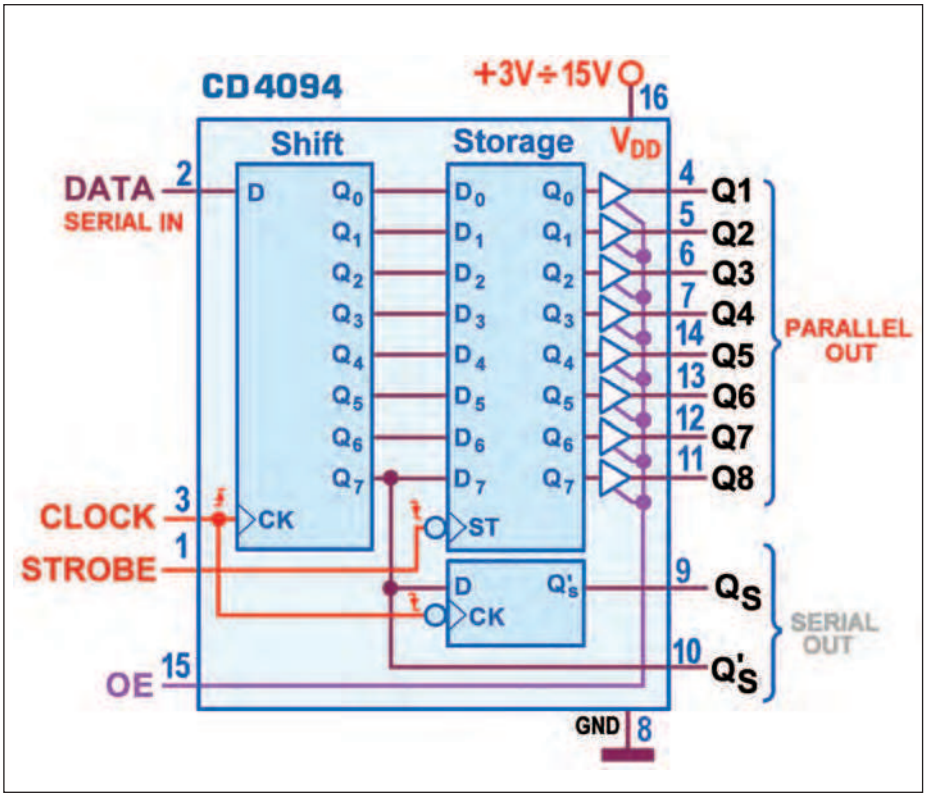


Figura 26: 8-Bit Shift Register SIPO with Output Latches, 4094: schema pratico.

Questo componente è particolarmente utile nelle operazioni di conversione serie-parallelo, magari con il 74LS674 (16-Bit PISO Shift Register), con cui fa coppia; la potenza massima dissipata è di 400 mW; la corrente massima I_{OL} assorbita in uscita è di 8 mA (per le 18 *Parallel Outputs*) e 24 mA (per la *Serial Output* Q_{15}'); i ritardi di propagazione massimi (con carico di 15pF/2000ohm) previsti tra i

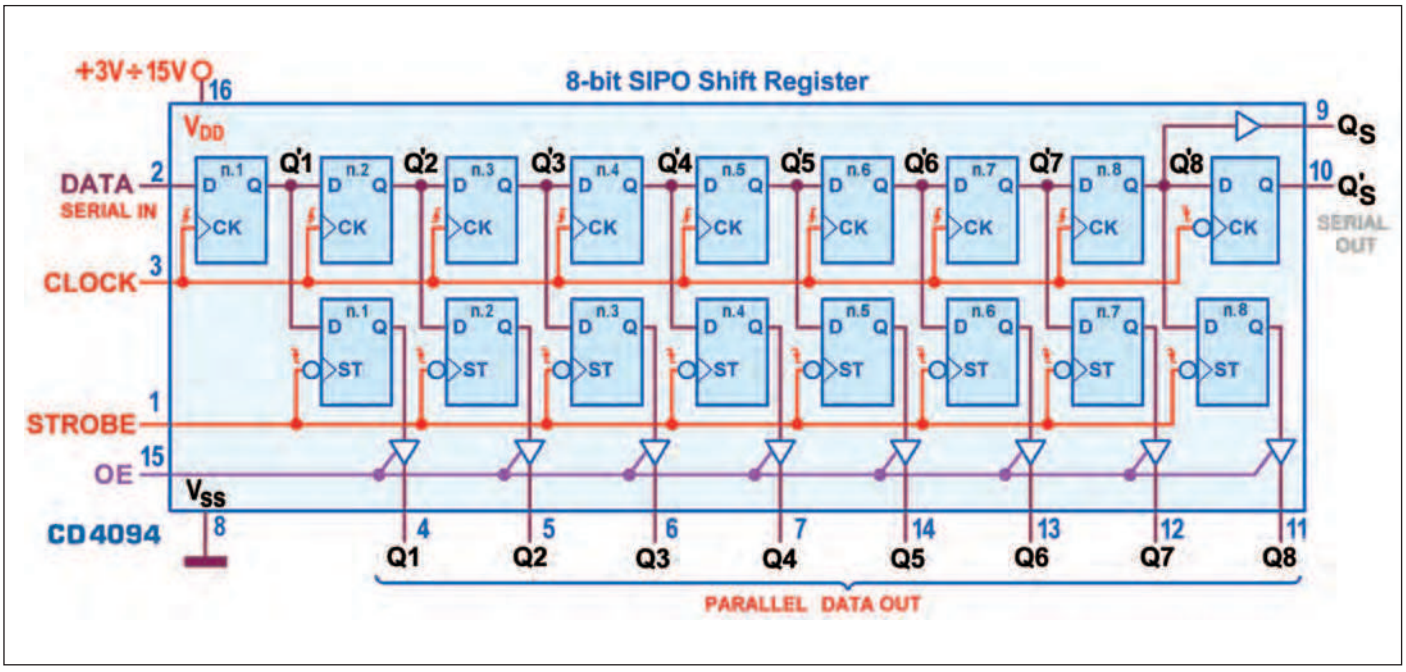


Figura 25: 8-Bit Shift Register SIPO with Output Latches, 4094: schema funzionale.

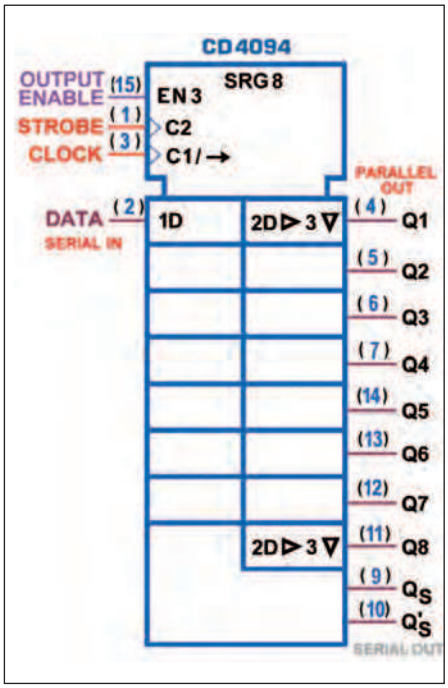


Figura 27: 8-Bit Shift Register SIPO with Output Latches, 4094: simbolo logico ANSI/IEEE Std. 91-1984.

vari segnali e le uscita parallele, da Y_{00} a Y_{15} , variano da 40 a 45 ns (per entrambe le transizioni), come quelli (con carico di 45pF/667ohm) tra gli stessi segnali e l'uscita seriale Q_{15}'; il valore minimo di Setup Time vale t_S=35 ns (per i controlli MODE/STRCK, CS e R/W) e 20 ns (per gli ingressi SER/Q15 e P_{00}-P_{15}), mentre quello di Hold Time è trascurabile. La frequenza massima di lavoro è di 20 MHz.

REGISTRI A SCORRIMENTO SIPO (CMOS): 4015, 4094

Il 4015 (un Dual 4-Bit SIPO Static Shift Register) contiene due Registri a Scorrimento a 4 bit, risultando il più utilizzato della famiglia CMOS per realizzare un semplice *SIPO*; la figura 20 mostra il suo pinout.

Dallo schema funzionale della **figura 21** si nota che ciascuno dei 2 Shift Register dispone della propria linea di Clock (attivo sul fronte salita), di Clear (attiva alta, cioè la cui azione, prioritaria, azzera tutte le uscite se forzata a 1), di *Serial Input* e di *Parallel Output*; l'uscita Q4 del quarto stadio è a disposizione come *Serial Output* per consentire la messa in cascata delle due metà e trasformare il componente in uno *Shift Register* a 8 bit, come il 74LS164, suo omologo in ambito TTL.

La figura 22 propone lo schema pratico con la Tabella di verità, che conferma la descrizione di questo oggetto, e la figura 23 mostra il simbolo logico previsto dallo standard IEEE: ogni considerazione sulle notazioni utilizzate è ormai nota e riconducibile a quanto già detto.

Come per gli altri componenti della famiglia CMOS la potenza dissipata dal 4015 è trascurabile mentre i tempi caratteristici (rilevati con carico di 200kohm/50pF) sono piuttosto elevati: il ritardo di propagazione, t_{PHL} e t_{PLH}, è al massimo di 350 ns (V_{DD}=5V), 160 ns (V_{DD}=10V) e 120 ns (V_{DD}=15V); sempre con riferimento alle tre tensioni di alimentazione, il valore minimo di Setup Time t_S vale 50 ns, 20 ns e 15 ns. La frequenza massima di lavoro è di 3,5 MHz, 8 MHz e 11 MHz.

L'altro componente della famiglia CMOS è il 4094, un 8-Bit SIPO Shift Register with Output Latches; la figura 24 mostra il suo pinout.

Si tratta di un componente logicamente riconducibile al TTL 74LS595, come esso caratterizzato dal fatto di contenere uno *Shift Register* a 8 bit di tipo *SIPO* con uscite collegate internamente agli ingressi di uno *Storage Register* di pari capacità; i bit forniti sull'ingresso seriale (*Serial Input*, Data) sono fatti scorrere negli 8 stadi dello Shift dopo ogni fronte di salita della linea di Clock, fino ad uscire dalla *Serial Output* Qs, utilizzabile per porre dispositivi dello stesso tipo in cascata.

Con riferimento allo schema funzionale di **figura 25** si notano però alcune differenze: a) il Registro *SIPO* è non dotato di linea di Reset; b) gli elementi di memoria della sezione di stoccaggio si comportano come Flip-Flops D-Latch: quando il comando *Strobe* è a 1, l'informazione presente sulle uscite interne del Registro a Scorrimento è lasciata passare sui buffer 3-state collegati alle sue uscite (ed eventualmente sulle *Parallel Outputs*, se OE=1), provvedendo a memorizzarle solo in presenza di un fronte di discesa di *Strobe*; il valore a monte dei buffer 3-state è mantenuto invariato per tutto il tempo durante il quale *Strobe*=0; c) la *Serial Output* Qs, è disponibile anche a valle di un "mezzo stadio" chiamato a memorizzare (contrariamente agli altri stadi) sul fronte di discesa del Clock: in questo modo il segnale presente sull'uscita dell'ultimo Flip-Flop viene ritardato di mezzo periodo di Clock, ritardando dello stesso tempo l'azione di shift sull'eventuale intergrato posto a valle, per facilitare la messa in cascata degli stadi quando (tra un integrato e il successivo) si verificano condizioni di carico onerose o tempi di commutazione critici.

La **figura 26** propone lo schema pratico; se la linea di *Output Enable* è forzata a 0 le *Parallel Output* sono poste in alta impedenza: questo rende il componente utile nelle strutture a bus.

La **figura 27** mostra il simbolo logico previsto dallo standard IEEE: la linea di Clock C1 impone lo spostamento verso destra (indicato dalla barra / e dalla freccia) dei dati in ingresso 1D (suggerito dal prefisso 1, riferito alla parte sinistra, Registro a Scorrimento, della parte bassa del simbolo). Lo Strobe C2 influisce invece sui dati 2D in ingresso al registro di uscita (riferito alla parte destra in basso, nel simbolo). Il prefisso 3 che precede il triangolo invertito (indicativo di uscite 3-state) sottolinea l'influenza su di essi da parte di EN3.

Come per gli altri componenti della famiglia CMOS la potenza dissipata dal 4015 è trascurabile mentre i tempi caratteristici (rilevati con carico di 200kohm/50pF) sono molto elevati: il ritardo di propagazione massimo (t_{PHL} e t_{PLH}) tra Clock e uscita seriale è di 600 ns (V_{DD}=5V), 250 ns (V_{DD}=10V) e 190 ns (V_{DD}=15V) e (sempre con riferimento alle tre tensioni di alimentazione) di ben di 840 ns, 390 ns e 270 ns tra Clock e uscita parallele; il valore minimo di Setup Time t_S vale 80 ns, 40 ns e 20 ns. La frequenza massima di lavoro è di 3 MHz, 6 MHz e 8 MHz. ❑

CODICE MIP 2805095

Hai un quesito di elettronica?
Vuoi comunicare con altri appassionati?
Vuoi condividere le tue esperienze?
Tutto questo su IEforum!
www.ieforum.it
CODICE MIP 2814094

di ANTONIO GIANNICO

ESPERIMENTI con i CONTATORI

I contatori sono dispositivi digitali in grado di effettuare il conteggio di impulsi applicati in ingresso. A seconda dei modelli, possono essere sincroni o asincroni, contare in avanti o all'indietro, essere dotati di ingressi di reset e di enable, essere presettabili o meno e utilizzabili in modalità binaria e/o BCD. Semplici esperimenti pratici su bread-board e alcune simulazioni circuitali al calcolatore possono essere di notevole aiuto nell'acquisizione pratica di questi concetti

I *contatori* sono circuiti costruiti da uno o più *flip-flop* in grado di effettuare il conteggio degli impulsi applicati sull'ingresso di clock. In generale, possono essere circuiti di tipo sincrono o asincrono, in grado di effettuare conteggi in avanti (*up counter*) o all'indietro (*down counter*), decimali (*BCD*) o binari, eventualmente con ingressi e funzioni di *Preset*, *Clear* ed *Enable*. Comprenderne le caratteristiche e il funzionamento è fondamentale per un loro corretto impiego più che conoscere quale sia la loro reale configurazione interna e quali siano i metodi di sintesi da cui essa discenda. Per questo motivo è necessario, prima di tutto, precisare quali siano le differenze tra *contatori asincroni* e *contatori sincroni*.

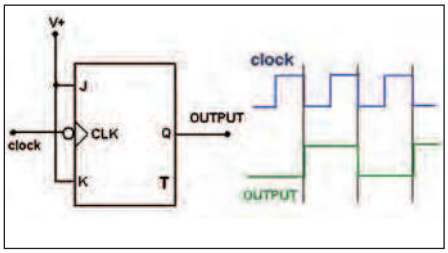


Figura 1: funzionamento di un flip-flop di tipo T.

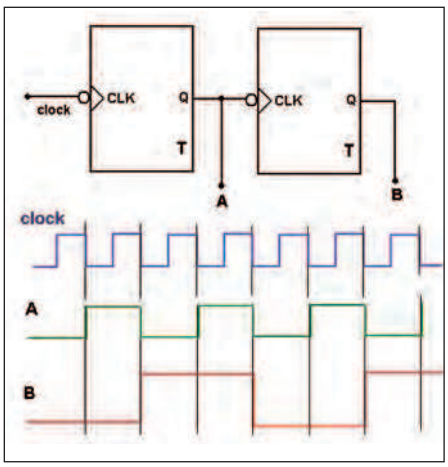


Figura 2: due flip-flop di tipo T connessi in cascata consentono di realizzare un contatore a due cifre binarie (per comprendere meglio l'affermazione si segua l'evoluzione temporale delle uscite A e B che rappresentano rispettivamente la cifra meno significativa e quella più significativa esprimenti lo stato del contatore).

I contatori asincroni sono costituiti da flip-flop che non commutano simultaneamente ma in successione poiché il clock del circuito non è applicato simultaneamente a tutti i flip-flop ma solo al primo di essi, il quale a sua volta, tramite un'apposita uscita lo propaga opportunamente al secondo, il quale in maniera analoga lo propaga al terzo e così via. Per questo motivo i contatori asincroni sono anche detti *ripple counter* o *contatori a riporto*. Ovviamente, se il numero di bit (e quindi di flip-flop) del contatore è n, 2^n è il numero massimo di stati in cui il contatore può trovarsi; tale valore è detto *modulo* del contatore. I contatori *BCD* (*Binary Coded Decimal*) sono in realtà caratterizzati da un numero di stati e quindi da un modulo inferiore rispetto a 2^n. Infatti il BCD è un sistema di codifica numerico che esprime le cifre da 0 a 9 del sistema decimale mediante l'impiego di 4 bit, tra-

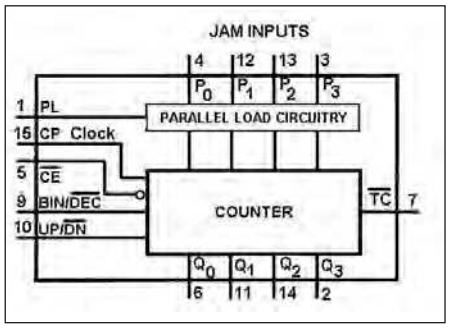


Figura 3: schematizzazione logica generale, package e pin-out del contatore CMOS 4029. Si faccia riferimento alla tabella 1 per dettagli sulla funzione dei diversi pin [1].

scurando le combinazioni che vanno da 10 a 15. Per esprimere un numero decimale con più cifre in codice BCD si codifica in pratica ciascuna delle cifre del numero decimale in forma binaria con una quaterna di bit (0→0000, 1→0001, 2→0010, 3→0011, 4→0100, 5→0101, 6→0110, 7→0111, 8→1000, 9→1001). In generale è possibile aumentare il modulo di un contatore asincrono collegando più contatori asincroni in cascata. Il limite principale di una tale connessione sta ancora una volta nei ritardi di propagazione che aumentano proporzionalmente al numero di stadi componenti il circuito stesso. Questo aspetto limita la frequenza di clock massima alla quale il contatore è in grado di operare.

Tra le caratteristiche più importanti di molti circuiti contatori vi è la programmabilità. Si tratta della caratteristica di questi componenti di essere presettabili, cioè della possibilità di precaricare uno stato iniziale. Il caricamento consiste nel trasferire sui pin di uscita una configurazione di altrettanti bit di ingresso di carico (Load) in maniera sincrona o asincrona dal clock a seconda del particolare modello di contatore. Tutti i bit della configurazione di carico sono in ogni caso trasferiti sulle uscite contemporaneamente, motivo per cui si parla di *caricamento parallelo* (*Parallel Load*).

Altre caratteristiche funzionali particolari sono conferite ai contatori dagli eventuali ingressi di *Reset* (o *Clear*) che consentono di impartire al circuito contatore un comando di azzeramento. Anche in questo caso il reset può avvenire in maniera sincrona con un impulso di clock oppure in maniera asincrona da esso. Nei paragrafi che seguono mostreremo alcuni esperimenti pratici su bread-board e alcune simulazioni circuitali relative a contatori sia CMOS che TTL. Tra essi prenderemo in considerazione, in particolare, i contatori CMOS 4029 e 4510 e il contatore TTL 7490.

Queste dimostrazioni, condotte secondo un approccio semplice, pratico e diretto, dovrebbero essere sufficienti a mettere il lettore nelle condizioni di comprendere il funzionamento della maggior parte dei contatori TTL e CMOS presenti in commercio. Questo è infatti il principale intento con cui è stato scritto il presente articolo. Nonostante ciò l'articolo è al tempo stesso una breve guida alla realizzazione pratica dei circuiti contatori TTL e CMOS più comuni. Anche per questo motivo, i circuiti presentati sono stati realizzati su bread-board o simulati con il simulatore Circuit Maker.

Quasi tutti i circuiti contatori impiegano come blocco principale il flip-flop di tipo T. Un flip-flop di questo tipo infatti può anche essere visto come un contatore a un bit che conta da 0 ad 1. In realtà un FF T altro non è che un circuito che cambia stato a ogni impulso di clock **(figura 1)**. Collegando opportunamente tra loro due flip-flop di tipo T è possibile ottenere un contatore a due bit in modalità asincrona **(figura 2)**. Per comprendere meglio l'affermazione è sufficiente seguire e analizzare in **figura 2** l'evoluzione temporale delle uscite A e B che rappresentano rispettivamente la cifra meno significativa e quella più significativa esprimenti lo stato del contatore. Come è facile verificare, il contatore transita attraverso i seguenti stati: 00→01→10→11→00→...

Questa logica può essere generalizzata aumentando opportunamente il numero di bit del contatore. In ogni caso, i contatori più comuni presentano tipicamente quat-

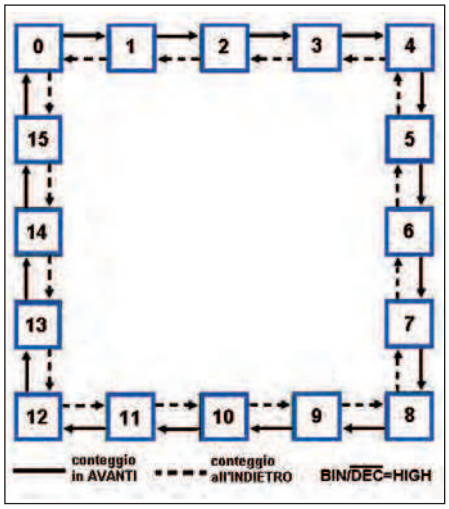


Figura 4: diagramma degli stati del contatore 4029 utilizzato in modalità binaria (BIN/DEC="1") [1].

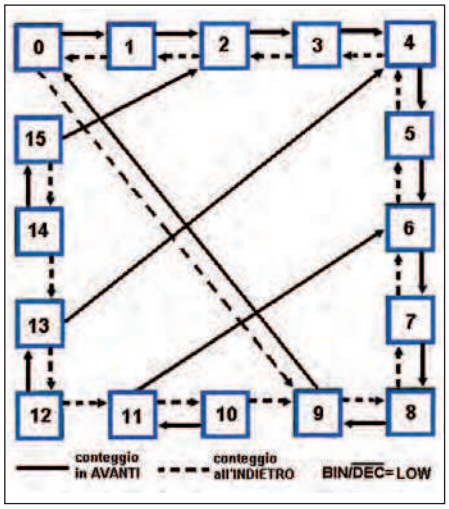


Figura 5: diagramma degli stati del contatore 4029 utilizzato in modalità BCD (BIN/DEC="0") [1].

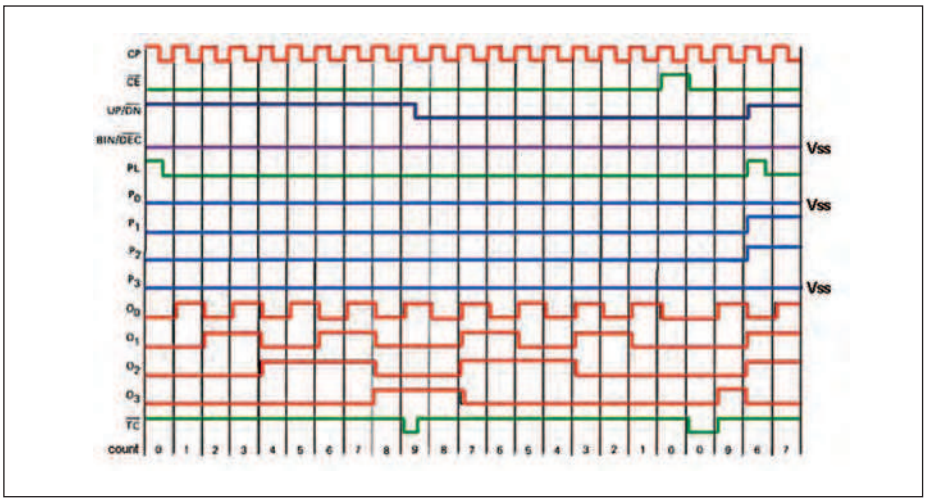


Figura 6: diagramma temporale del contatore 4029 (modalità BCD) [1].

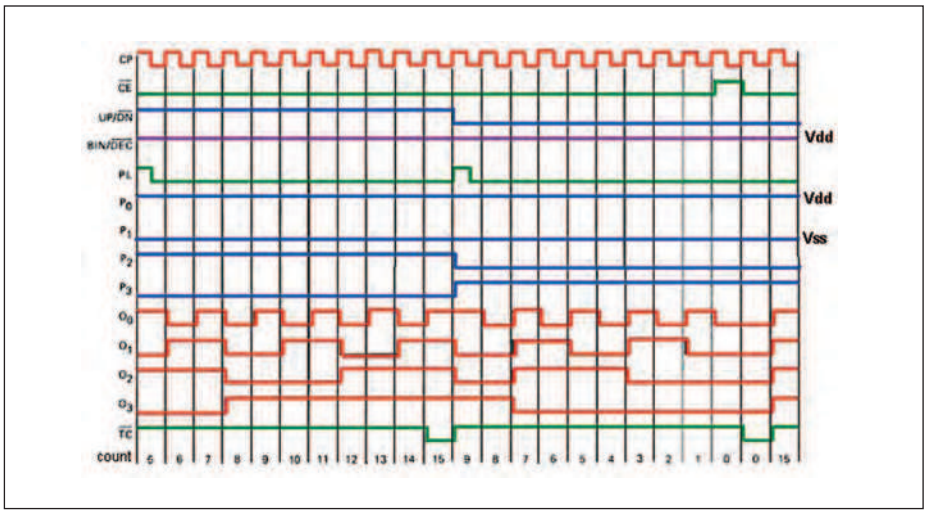


Figura 7: diagramma temporale del contatore 4029 (modalità Binaria) [1].

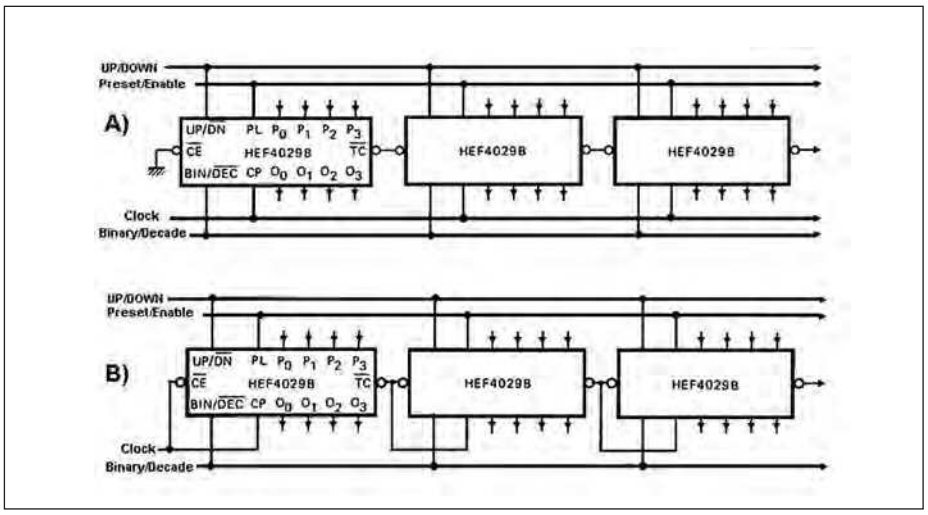


Figura 8: connessione in cascata di più contatori 4029 (A-Parallel Cloking; B-Ripple Clocking) [1].

tro bit e pertanto contengono quattro flip-flop. Il conteggio passa da 0 ad 1, poi a 2 e così via fino a15, stato raggiunto il quale un successivo impulso di clock determina il reinizio del ciclo di funzionamento dallo stato 0. Un conteggio di questo tipo è binario in quanto evolve attraverso tutti gli stati binari possibili con quattro bit. I contatori BCD funzionano esattamente allo stesso modo, tuttavia interrompono allo stato 9 il conteggio, ripartendo al successivo impulso di clock dallo stato 0. Il contatore è per questo internamente modificato rispetto a quello binario in maniera tale che un circuito apposito rilevi lo stato 9 e riazzeri il contatore al verificarsi del successivo impulso di clock. I contatori BCD sono in realtà quelli più diffusi poiché consentono di effettuare conteggi in base 10 (conteggi composti da unità, decine, centinaia e così via) e poiché connessi a opportuni decoder BCD-7segmenti consentono di pilotare display a 7 segmenti per la visualizzazione del valore su un numero di diplay a 7 segmenti pari al numero di cifre che compone il conteggio stesso.

In commercio esistono integrati contatori che possono essere impostati in maniera tale da contare solo in avanti oppure sia in avanti sia all'indietro, in modalità binaria o in modalità BCD o in entrambe le modalità. Evidentemente, affinché sia possibile gestire queste modalità di funzionamento è necessario che il chip presenti determinati pin di controllo. Quanto verrà esposto nel successivo paragrafo circa il contatore CMOS 4029 e successivamente ancora circa i contatori 4510 e TTL 7490 dovrebbe risultare sufficiente a chiarire ulteriormente quanto detto fino a questo momento sui contatori.

IL CONTATORE 4029

Il CD4029 è un *contatore CMOS sincrono presettabile, edge triggered* (commuta di stato sul fronte di salita del clock) *up/down a quattro bit* che può operare sia in modalità *binaria* che *BCD*. Esso integra la maggior parte delle funzioni che generalmente è possibile trovare in un contatore. Per questo motivo è opportuno analizzarlo in dettaglio. A tale proposito faremo riferimento alle **tabelle 1** e **2** e alle figure **3**, **4**, **5**, **6** e **7** cui sarà necessario prestare particolare attenzione. Nel corso dei paragrafi che seguiranno eseguiremo alcuni esperimenti pratici proprio su questo modello di contatore, vista la sua flessibilità operativa.

Molte delle cose che diremo rappresentano in ogni caso concetti ricorrenti per questo genere di componenti.

La prima impostazione necessaria in un contatore di questo tipo, per il suo corretto impiego, è quella relativa alla modalità operativa. Essa è fissata attraverso il livello di tensione che si applica sull'apposito ingresso *BIN/DEC* (*Binary/Decade control input*). Il valore logico "1" imposta il contatore come binario mentre il valore logico "0" lo imposta come decimale o, come si dice più correttamente, BCD. Mentre in configurazione binaria il contatore può percorrere 16 stati, corrispondenti all'insieme di tutte le possibili configurazioni dei quattro bit della porta di uscita (da "0000" ad "1111"), in configurazione BCD può percorrere i soli primi dieci stati (da "0000" a "1001"). La modalità BCD è quella che consente di implementare circuiti contatori dotati di peso in base dieci, caratterizzati cioè da unità, decine, centinaia e così via. Appositi decoder BCD-7segmenti consentono la rappresentazione delle singole cifre BCD (cifre da "0" a "9" espresse in binario con quattro bit) su display a 7 segmenti.

L'ingresso *UP/DN* (acronimo di "*up/down*") serve per impostare il verso ciclico del contatore. In altre parole, l'impostazione

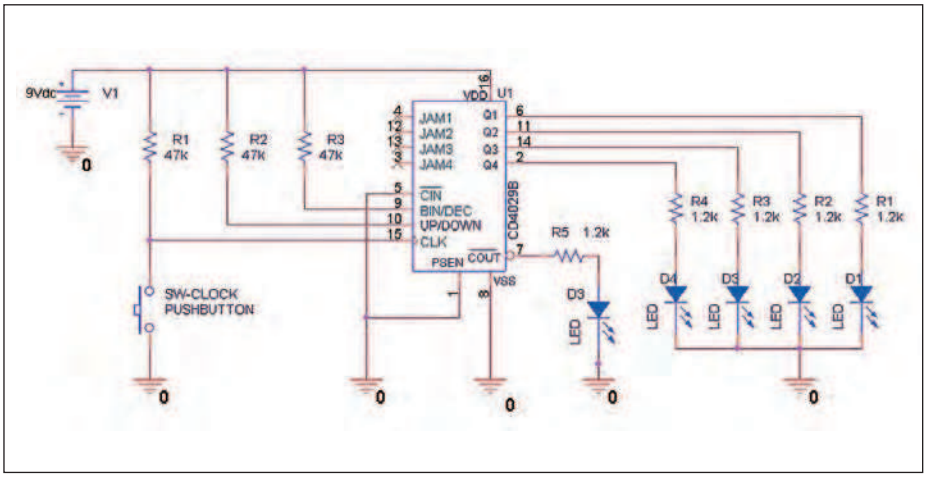


Figura 9: semplice schema circuitale per la verifica del funzionamento di un contatore CMOS 4029.

del pin a livello logico "1" comporta una modalità di conteggio "*UP*" cioè in avanti e pertanto in senso crescente, mentre l'impostazione a livello logico "0" comporta una modalità di conteggio "*DOWN*" cioè all'indietro e pertanto in senso decrescente.

Analogamente, l'ingresso *PL* (Parallel Load input) altro non è che un "preset enable" cioè una abilitazione al preset dello stato del contatore. Se tale pin è portato a livello "1", il contatore è abilitato ad accettare dei livelli di preset posti su una porta di quattro bit in ingresso (Parallel Data Inputs, normalmente indicata con $P_0P_1P_2P_3$ dove P sta per "Preset" oppure con $J_0J_1J_2J_3$ dove J sta per JAM Inputs). L'intera parola così formata viene trasferita in modalità parallela sui pin di uscita e costituisce il nuovo stato di partenza del contatore. Tale comando determina quindi il preset dell'uscita che agisce in maniera asincrona rispetto al clock. Il contatore avanza nel suo conteggio (in avanti o all'indietro a seconda della modalità impostata) in corrispondenza del fronte di salita positivo del clock (edge trigger), a patto che gli ingessi di preset non siano abilitati (PL=0), e che sia invece abilitato il conteggio (CE=0). A tale proposito, è opportuno notare la logica negata degli ingressi CE e PL. Il pin CE (attivo basso) è il pin di abilitazione al conteggio. CE sta infatti per "*Count Enable*" o, in modo equivalente, per "*Clock Enable*". Questo pin è spesso anche detto "Carry In" per motivi che appariranno evidenti nel corso dell'esposizione. Evidentemente, è sufficiente che uno tra i pin di abilitazione del preset (PL) e di enable (CE) siano a livello alto perché l'evolvere del conteggio risulti inibito.

Il pin *TC* (Terminal Count) rappresenta il cosiddetto "Carry Out" poiché segnala il raggiungimento dello stato di valore massimo in modalità di conteggio in avanti e quello di valore più basso in modalità di conteggio all'indietro. Durante il conteggio, esso è normalmente a livello alto "1", mentre si porta a livello basso quando il conteggio raggiunge il valore finale. Il motivo per cui si parla anche di pin di "Carry Out" apparirà evidente nel corso delle simulazioni che proporremo nei prossimi paragrafi e nelle quali lo stesso sarà utilizzato per la propagazione del clock nella realizzazione di circuiti contatori asincroni in cascata. Ovviamente, come qualunque contatore, il 4029 è dotato di un apposito pin di clock indicato con *CP,* acronimo di Clock Pulse o con *CK,* acronimo più semplicemente di clock. Ovviamente, essendo il *TC* (Terminal Count) attivo basso, esso risulterà a livello alto durante il conteggio e si porterà a livello basso alla fine dello stesso cioè quando si raggiungerà la configurazione di valore massimo in *up mode* e quella di valore minimo in *down mode*.

È importante sottolineare che i valori presenti sui pin $P_0P_1P_2P_3$ vengono in maniera asincrona caricati sulle uscite $Q_0Q_1Q_2Q_3$ solo se l'abilitazione del preset è a livello alto (PL=1) ma in ogni caso indipendentemente dal clock. Il contatore avanza nel suo conteggio se è impostato in modalità up-mode o regredisce se è impostato in modalità down mode nel momento i cui il livello di tensione sul pin CP transita da valore basso a valore alto (ammesso che in questi frangenti i pin CE e PL si trovino a livello basso).

PER *approfondire...*

[1] HEF4029B MSI Synchronous up/down counter, binary/decade counter (January 1995, Philips Semiconductors)

[2] Il Digitale - R. Giometti, F. Frascari 3a Ed. - Calderini Editore. Cap. 3 - Elementi di logica Sequenziale -Esempi ed applicazioni dei contatori integrati

[3] HEF4017B MSI-5-stage Johnson counter (January 1995, Philips Semiconductors)

[4] HCC/HCF4017B HCC/HCF4022B Counters/Dividers (1994 SGS-Thomson Microelectronics)

[5] CD4029BM/CD4029BC-Presettable Binary/Decade Up/Down Counter (February 1988, National Semiconductors)

[6] SN54/74LS90, SN54/74LS92, SN54/74LS93 - Decade Counter; Divide-By-Twelve Counter; 4-Bit Binary Counter (fonte Motorola)

[7] SN54/74LS160A, SN54/74LS161A, SN54/74LS162A, SN54/74LS163A- BCD Decade Counters/ 4-Bit Binary Counters (fonte Motorola)

[8] SN54/74LS168, SN54/74LS169 - BCD Decade/Modulo, 16 Binary Synchronous Bi-Directional Counters (fonte Motorola)

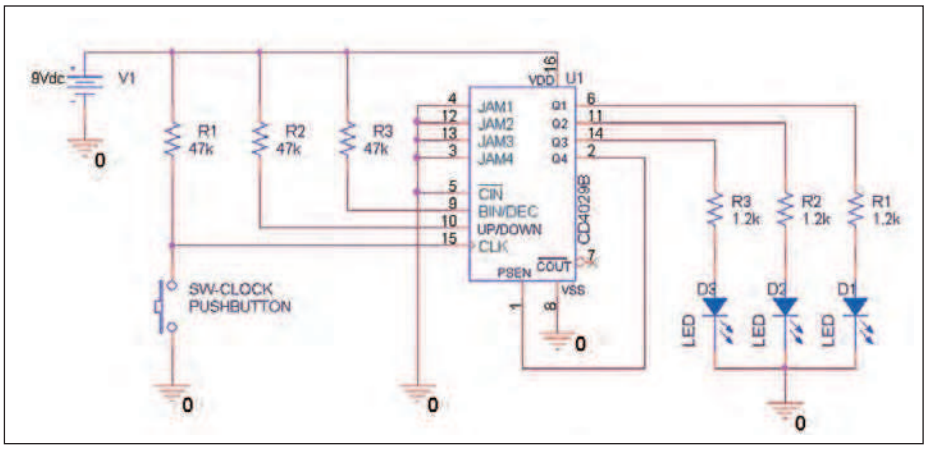


Figura 10: schema circuitale per la verifica del funzionamento di un contatore CMOS 4029 con azzeramento del conteggio al raggiungimento dello stato "7" attraverso il riporto dell'uscita Q4 sul pin 1 (Preset Enable).

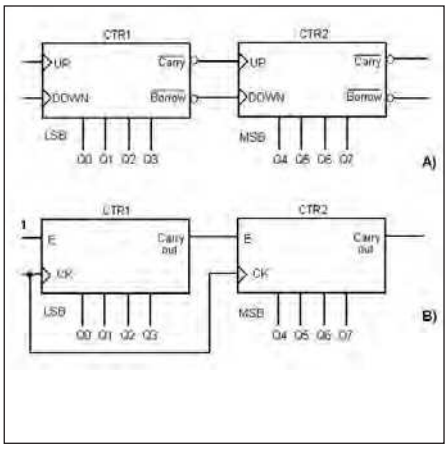


Figura 12: connessione di contatori in cascata: a) asincrona; b) sincrona.

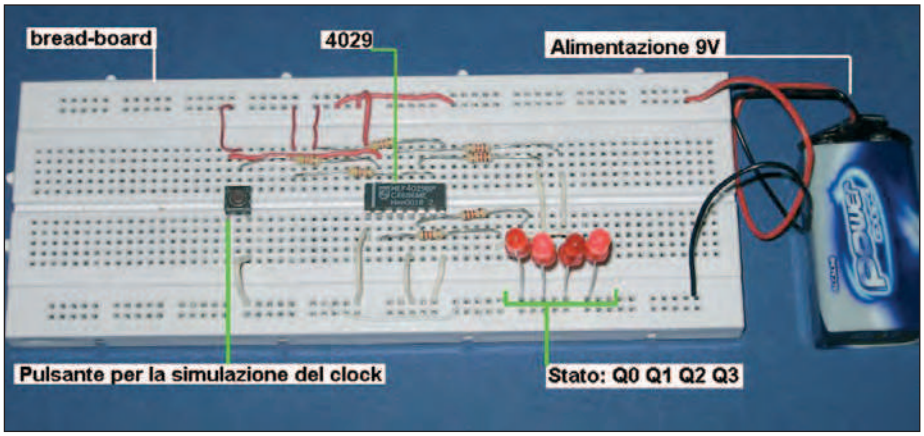


Figura 11: implementazione su bread-board dello schema circuitale di figura 9.

Il funzionamento del contatore è riassunto sinteticamente attraverso le **tabelle 1** e **2** e il pin-out di **figura 3** oppure attraverso appositi diagrammi a stati che ne esprimono il funzionamento, rispettivamente in modalità binaria **(figura 4)** e in modalità BCD **(figura 5)** o ancora attraverso opportuni diagrammi temporali **(figure 6** e **7)**. Quanto illustrato, sebbene riferito al contatore CMOS 4029, dovrebbe risultare utile a comprendere le funzioni associate ai diversi pin di qualunque integrato contatore.

APPLICAZIONI

Come è facile immaginare, le applicazioni dei contatori sono molteplici in quanto molteplici sono le attività nelle quali è necessario eseguire operazioni di conteggio, specie se di tipo automatico. Oltre alle applicazioni classiche di conteggio in avanti o all'indietro vi sono quelle in cui si necessita di conversione analogico-digitale oppure di divisione di frequenza di un clock. Contatori alla rovescia, orologi digitali, conteggio elettronico nell'automazione (si pensi a un semplice contapezzi elettronico), conteggio accessi sono tutte applicazioni pratiche dei circuiti contatori piuttosto ricorrenti. Ovviamente, nelle applicazioni reali un singolo contatore spesso non è sufficiente a svolgere la relativa funzione per cui si implementano contatori a più cifre impiegando più contatori BCD opportunamente connessi in cascata tra loro.

In **figura 8** è schematizzato questo tipo di implementazione con riferimento al contatore 4029 (8A-parallel clocking ovvero connessione con clock comune a tutti i contatori; 8B-ripple clocking ovvero configurazione con clock propagato da un contatore al successivo attraverso il pin TC, Terminal Count).

ESPERIMENTI PRATICI CON I CONTATORI

Il modo più semplice di familiarizzare con questi componenti e sperimentare i concetti fin qui esposti è quello di implementarne i circuiti applicativi più ricorrenti su bread-board o all'interno di un simulatore circuitale. Si tratta infatti di circuiti il cui funzionamento può essere spesso esaminato con pochi espedienti pratici. Per esempio è possibile impiegare semplici diodi LED posti sui rispettivi pin di uscita al fine di osservare lo stato del contatore e la sua evoluzione nel tempo. Il clock deve in questo caso risultare sufficientemente lento tanto da consentire l'osservazione visiva dello stato e il passaggio da uno stato all'altro. Il clock potrebbe inoltre facilmente essere ottenuto anche senza disporre di un generatore di funzioni; per esempio potrebbe essere implementato con un timer 555. Una sorgente di alimentazione e qualche dip-switch o jumper per l'impostazione dei pin di controllo sono sufficienti a completare il set minimo di componenti necessari per i primi esperimenti. Volendo evitare la necessità di disporre o costruire un clock è anche possibile impiegare un pulsante che attraverso un pull-up simuli la discesa e la salita del segnale sul relativo pin. Avere a disposizione un generatore di segnali è quindi non strettamente necessario per l'esecuzione di semplici esperimenti funzionali e la stessa cosa vale per l'oscilloscopio. Al fine di eseguire gli esperimenti nel modo più semplice possibile è inoltre opportuno utilizzare un contatore CMOS che tipicamente è

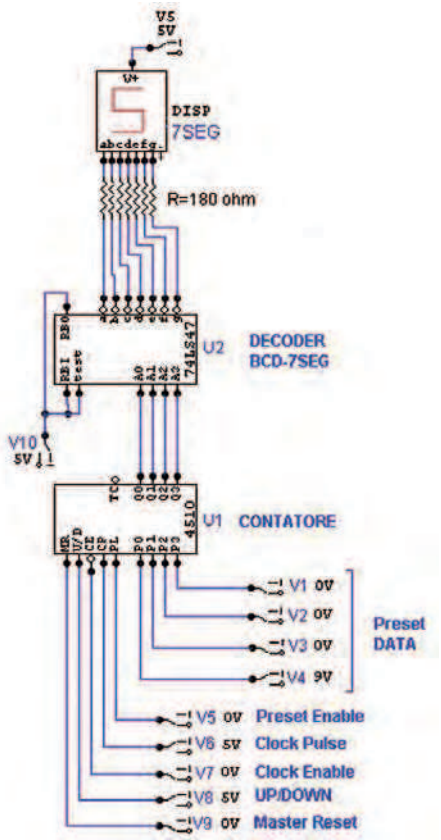


Figura 13: schema circuitale di un contatore a una cifra decimale (conteggio 0..9) basato sull'impiego del CMOS sincrono 4510 e visualizzazione su display a 7 segmenti ad anodo comune (si notino il decoder BCD-7segmenti 7447 e le predisposizioni per il pilotaggio dei pin di controllo).

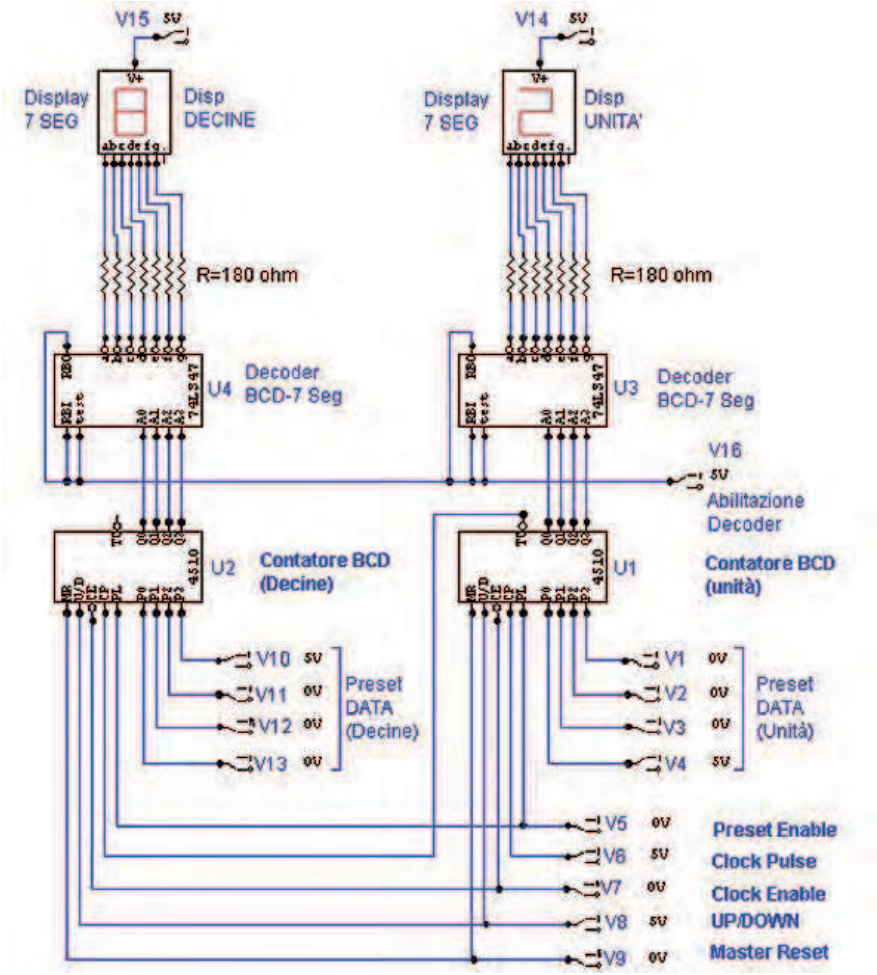


Figura 14: schema circuitale di un contatore a due cifre decimali (conteggio 0..99) basato sull'impiego di due contatori CMOS 4510 e visualizzazione su display a 7 segmenti ad anodo comune, tramite l'impiego di altrettanti decoder BCD-7segmenti 7447. A ciascuna cifra decimale corrisponde un contatore, un decoder e un display a 7 segmenti. Il circuito costituisce la naturale generalizzazione di quello di figura 13 e può a sua volta essere ulteriormente esteso per generalizzazione all'implementazione di un contatore a più cifre.

in grado di accettare tensioni di alimentazioni comprese tra 3 V e 15 V. Questo consente di alimentare il circuito contatore implementato su bread-board con una comune batteria dello stesso tipo di quelle utilizzate per i telecomandi **(figura 11)**. Se negli esperimenti su bread-board che seguiranno non utilizzeremo contatori TTL è quindi solo per motivi di opportunità legate alla sorgente di alimentazione. Per questo motivo eseguiremo esperimenti su circuiti contatori TTL solo sotto forma di simulazioni circuitali. Questi ultimi riguarderanno in particolare il contatore TTL BCD a quattro bit 7490. Chi dovesse impiegarlo nei suoi esperimenti su bread-board deve ricordare che il suo pin-out è differente da quello abituale delle logiche TTL. Infatti, invece di presentare, come normalmente ci si aspetta, la massa sul pin 7 e l'alimentazione sul pin 14, presenta la massa sul pin 10 e l'alimentazione positiva sul pin 5.

ESPERIMENTI SU BREAD-BOARD CON CONTATORE CD4029

Abbiamo già evidenziato che al fine di eseguire esperimenti sui contatori disponendo di una semplice bread-board, una pila da 9 V per l'alimentazione e una manciata di diodi, utilizzeremo come integrato contatore il CMOS 4029. Anche coloro che non dispongono di un vero laboratorio potranno in questo modo ripetere facilmente gli esperimenti stessi. Il clock verrà simulato manualmente attraverso un pulsante e un pull-up. Unico limite di questo approccio è costituito dagli effetti di rimbalzo che potrebbero manifestarsi in corrispondenza della pressione o del rilascio del pulsante e che, nel corso di alcune transizioni di clock, potrebbero falsare il comportamento atteso.
Il circuito di riferimento è quello di **figura 9** mentre la relativa implementazione su bread-board è mostrata in **figura 11**. Esso consente di sperimentare facilmente l'effetto del clock e quello del comando UP/DOWN. Per effettuare un conteggio in modalità ascendente sarà sufficiente porre un livello logico "1" sull'ingresso di controllo U/D mentre per ottenere un conteggio in modalità discendente sarà

sufficiente porre sullo stesso pin un livello logico "0". Analogamente sarà possibile verificare lo stato dell'uscita C/O (pin 7-Carry Out). Se il contatore viene impiegato in modalità ascendente, questo pin si attiva quando il conteggio arriva al valore limite superiore, mentre se è impiegato in modalità discendente si attiva quando il conteggio raggiunge il livello limite inferiore. Il concetto di pin che si attiva va interpretato correttamente. Il simbolo di "negato" sul pin C/O indica che l'attivazione va intesa a livello basso, pertanto al raggiungimento dell'evento il pin commuterà da livello alto a livello basso. Aggiungendo un LED su questo pin sarà possibile osservarne visivamente l'attivazione. Un altro esperimento semplice da effettuare mediante questo circuito è quello relativo al controllo dell'enable che può essere impiegato come comando di Halt (nel caso del 4029 questa funzione è resa disponibile dal pin 5, comunemente indicato con l'acronimo CE-Clock Enable o, in modo equivalente, con l'acronimo CI-Clock IN). Trattandosi di un pin negato è attivo a livello basso, pertanto se questo pin è posto a livello logico "0" il contatore conta normalmente, mentre se è posto a livello logico 1 le uscite vengono bloccate come se il clock fosse congelato. Il circuito di **figura 9** è sufficiente a sperimentare e verificare direttamente tutte queste funzionalità per il contatore CMOS 4029. In **figura 11** ne è mostrata l'implementazione fisica su bread-board. Modificando leggermente il circuito (agendo semplicemente sullo stato del pin B/D che serve a impostare la modalità di conteggio) è inoltre possibile sperimentare la differenza tra conteggio binario e conteggio BCD. Trattandosi di un contatore a quattro bit potrà contare da 0000 a 1111, cioè tra 0 e 15 se configurato in modalità binaria, e da 0000 a 1001, cioè da 0 a 9, in modalità decimale (BCD). Anche le funzioni di preset possono essere sperimentate. Il comando su cui agire è quello del pin 1 (preset, acronimo PL, PSEN o PE). Se questo pin si trova a livello basso il contatore si comporta normalmente contando nella direzione imposta dall'ingresso UP/DOWN, se è invece portato a livello alto il contatore smette di contare e trasferisce direttamente e asincronicamente dal clock i valori binari presenti sulla porta di preset (porta indicata con l'acronimo P o J) sui quattro bit di uscita (porta indicata con l'acronimo O o Q). Una funzionalità di questo tipo può essere utilizzata per modificare il ciclo di conteggio. Per esempio, se si desidera che il contatore conti da 0 a 7 collegheremo l'uscita Q4 al piedino di abilitazione del preset (PSEN) ponendo una quaterna 0000 sugli ingressi di preset $J_1J_2J_3J_4$ **(figura 10)**. Seguendo le accensioni e gli spegnimenti dei LED sulla porta di uscita Q è possibile verificare in ciascun istante quale sia lo stato in cui si trova il contatore. Modificando opportunamente il circuito di **figura 9** o quello di **figura 10** in maniera tale da portare sul pin UP/DOWN un livello alto o basso è possibile verificare il conteggio in avanti e quello all'indietro e, agendo analogamente sul pin BCD/DEC, è possibile osservare il passaggio della modalità di funzionamento da quella binaria a quella BCD o viceversa. Aggiungendo un ulteriore LED sul pin 7 (Terminal Count) è possibile verificare il raggiungimento del-

TABELLA 1

PIN	ACRONIMO	SIGNIFICATO / FUNZIONE
1	PL (o PE)	Parallel Load input (Preset Enable)
4-12-13-3	da P0 a P3	Parallel Data inputs
9	BIN/DEC	Binary/Decade control input
10	UP/DN	Up/Down control input
5	CE	Count Enable input (active LOW). Anche detto Carry IN
15	CP (o CK)	Clock Pulse o Clock Input (edge triggered)
6-11-14-2	da Q0 a Q3	Buffered Parallel output
7	TC	Terminal Count output (active LOW). Anche detto Carry Out
16	VDD	Alimentazione positiva
8	VSS	Alimentazione negativa

Tabella 1: pin out del contatore integrato sincrono Edge Triggered presettabile CMOS 4029.

PER *approfondire...*

[9] SN54/74LS190, SN54/74LS191 - Presettable BCD/Decade Up/Down Counters Presettable 4-Bit Binary Up/Down Counters (Fonte Motorola)

[10] SN54/74LS192, SN54/74LS193 - Presettable BCD/Decade Up/Down Counter Presettable 4-Bit Binary Up/Down Counter (fonte Motorola)

[11] SN54/74LS290, SN54/74LS293 - Decade Counter; 4-Bit Binary Counter (fonte Motorola)

[12] SN54/74LS390 SN54/74LS393 - Dual Decade Counter; Dual 4-Stage Binary Counter (fonte Motorola)

[13] CD4510B, CD4516B Types - CMOS Presettable Up/Down Counters - Texas Instruments (www.datasheet)

[14] DM7490A - Decade and Binary Counters - August 1986, Revised March 2000 (www.fairchildsemi.com)

[15] SN5446A, '47A, '48, SN54LS47, 'LS48, 'LS49, SN7446A, '47A, '48, SN74LS47, 'LS48, 'LS49 BCD-To-Seven-Segment Decoders/Drivers - March 1974 - Revised March 1988 (Texas Instruments)

IQRF®

Smarter wireless. Simply.

IQRF Wireless Challenge

Design Contest

Mettiti alla prova su
www.iqrf.org/contest

Crea un nuovo progetto con la comunicazione wireless e vinci premi per un valore di 10000 USD

Premio in denaro 6000 USD

La registrazione parte il 1° Dicembre 2011

IQRF è una tecnologia completa per la connettività RF

- Cimentati con i moduli transceiver intelligenti TR-5xB con sistema operativo a bordo, e il supporto di IQMESH
- Costruisci la tua **rete Wireless Mesh con facilità**

Molto facile da implementare 868 MHz o 916 MHz 35 µA Rx!

Questo contest è stato organizzato con il patrocinio di:

- *IARIA – International Academy, Research, and Industry Association*
- *Ministry of Industry and Trade of the Czech Republic*

CODICE MIP 2814098

IQRF è distribuito in Italia da
Inware
Providing Innovation
www.inware.it

IARIA

MICRORISC

MINISTRY OF INDUSTRY AND TRADE

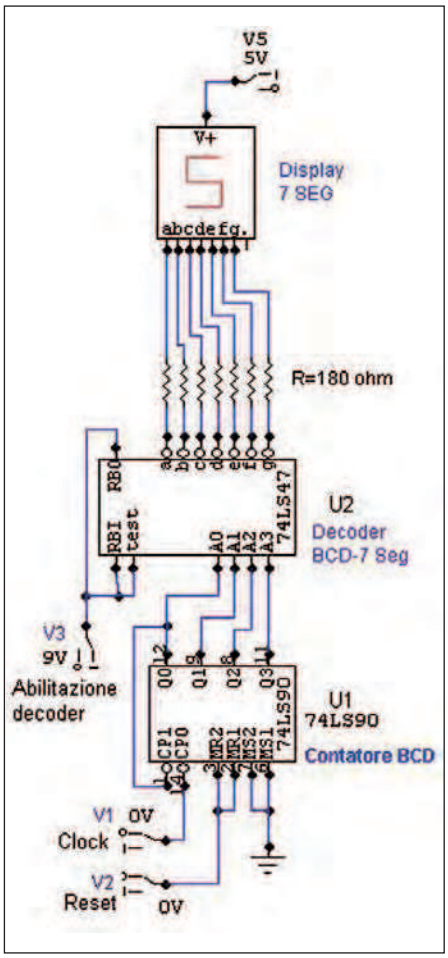


Figura 15: schema circuitale di contatore a una cifra decimale (conteggio 0..9) basato sull'impiego di un contatore TTL 7490 e visualizzazione su display a 7 segmenti ad anodo comune, tramite l'impiego di un decoder BCD-7segmenti 7447.

TABELLA 2

PL	BIN/DEC	UP/DN	CE	CP	FUNZIONAMENTO
H	X	X	X	X	Preset (Parallel Load) PnàQn
L	X	X	H	X	Nessuna variazione di stato
L	L	L	L		Conteggio BCD all'indietro
L	L	H	L		Conteggio BCD in avanti
L	H	L	L		Conteggio binario all'indietro
L	H	H	L		Conteggio binario in avanti

Tabella 2: funzionamento del contatore Up/Down presettabile 4029. Si faccia riferimento alla figura 3 per dettagli circa il pin-out [1].

TABELLA 3

SIGLA	TECNOLOGIA	CARATTERISTICHE
7490	TTL	Contatore a decade
7492	TTL	Contatore di modulo 12 a quattro bit
7493	TTL	Contatore binario a 4 bit
74290	TTL	Contatore equivalente al 7490 ma con diversa piedinatura
74293	TTL	Equivalente al 7493 a quattro bit con diversa piedinatura
74390	TTL	Contiene due contatori a quattro bit di tipo 74290
74393	TTL	Contiene 2 contatori a quattro bit del tipo 74293
4020	CMOS	Contatore a quattordici stadi
4024	CMOS	Contatore a 7 stadi
4040	CMOS	Contatore a 12 stadi

Tabella 3: principali contatori asincroni TTL e CMOS.

la fine del conteggio mentre, agendo sul pin PL e utilizzando un'opportuna configurazione di bit sui pin JAM1…4, è possibile eseguire ulteriori verifiche della funzionalità di preset.

ALTRI INTEGRATI CONTATORI

Il circuito integrato 4029 è realizzato in tecnologia CMOS. Esistono in commercio anche numerosi circuiti integrati in tecnologia TTL. È opportuno presentare, almeno per alcuni di essi e almeno per le applicazioni pratiche più comuni, alcune semplici simulazioni circuitali al calcolatore. A tale scopo faremo riferimento soprattutto all'integrato contatore TTL 7490. Si tratta di un contatore a quattro bit in grado di effettuare un conteggio BCD che si manifesta sui pin 12, 9, 8 e 7, corrispondenti rispettivamente alle cifre binarie Qa, Qb, Qc e Qd. Il contatore TTL funzionalmente più simile al CMOS 4029 è invece il TTL 74193 poiché come il 4029 può essere precaricato con la configurazione desiderata, può essere configurato per conteggio up o down e presenta un pin di segnalazione di fine conteggio (CO-Carry-Out) indicante il raggiungimento del valore massimo e uno di segnalazione del valore minimo corrispondente allo stato "0" (BO-Borrow). Un'altra particolare funzione integrata da questo chip è quella di Clear (CLR), che consente di effettuare una cancellazione (o meglio azzeramento) dell'uscita.

Altro contatore TTL degno di nota è il 74160, contatore BCD con funzioni di reset e di preset. È interessante osservare che si tratta di un contatore sincrono che consente quindi velocità di clock superiori a quelle tipiche dei contatori asincroni. Attraverso i datasheet, alcuni dei quali elencati all'interno del riquadro "Per approfondire" è possibile approfondire ulteriormente la conoscenza di questi dispositivi. Per completezza, in **tabella 3** riassumiamo i principali contatori asincroni disponibili in commercio, realizzati sia in tecnologia TTL sia CMOS. Quelli riportati in **tabella 3** sono in ogni caso contatori asincroni. Diversamente da questi, quelli sincroni, pur presentando le stesse funzionalità garantiscono prestazioni più elevate in termini di clock, caratteristica ottenuta a scapito di una maggiore complessità circuitale. Come i contatori asincroni, anche quelli sincroni possono essere binari o a de-

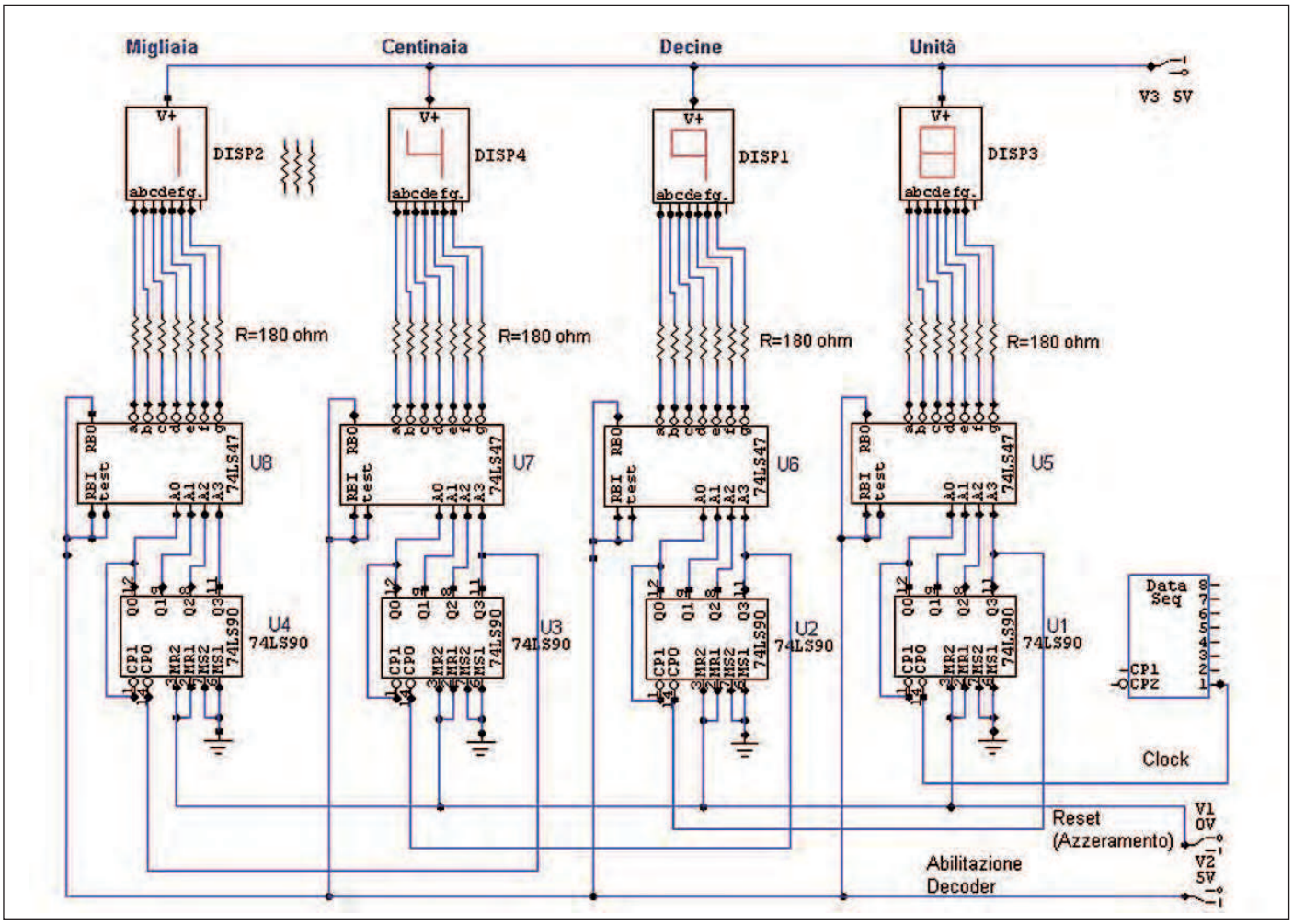


Figura 16: schema circuitale di contatore a quattro cifre decimali (conteggio 0..9999) basato sull'impiego di quattro contatori TTL 7490 e visualizzazione su display a 7 segmenti ad anodo comune, tramite l'impiego di decoder BCD-7segmenti 7447. A ciascuna cifra decimale corrisponde un contatore, un decoder e un display. Il circuito costituisce la naturale generalizzazione di quello di figura 15.

cade, cioè BCD. Questi ultimi (BCD) possono essere visti come particolari tipi di contatore a sequenza troncata dal momento che la sequenza degli stati viene troncata a un modulo pari a $M<2^n$. Anche i contatori sincroni possono essere connessi in cascata al fine di ottenere un contatore di modulo maggiore. Ancora una volta, connettendo opportunamente in cascata due o più contatori è possibile ottenere un contatore di modulo pari al prodotto dei moduli. A questo servono l'uscita di conteggio finale e l'ingresso di abilitazione di cui questi circuiti sono generalmente dotati. L'uscita di conteggio finale in particolare è normalmente indicata con l'acronimo *TC* (Terminal Count) oppure con *RCO* (Ripple Carry Output, uscita di propagazione del riporto o *Ripple Clock Out*, uscita di propagazione del clock) o ancora con *Carry Out* o *Carry*. L'ingresso di abilitazione al conteggio è invece normalmente indicato con l'acronimo *CTEN* (Count ENable) o con *E* (Enable) o con ENT o ENP. Si tratta in ogni caso di acronimi ricorrenti nell'àmbito dei circuiti contatori o a stati finiti. In **figura 12** è riportato lo schema di principio della connessione che consente di ottenere l'espansione e che fa riferimento in particolare al contatore integrato 74193. Quando il contatore conta in avanti, gli impulsi di clock sono applicati all'ingresso UP **(figura 12 a).** Quando si giunge all'ultimo stato (15) l'uscita Carry si porta a livello basso. Al successivo impulso di clock il Carry passa a livello alto nuovamente e questo fronte attiva il contatore CTR2 mentre il contatore CTR1 torna ad azzerare i suoi bit. Nel conteggio decrescente il clock è applicato all'ingresso DOWN del contatore CTR1; quando si giunge a valore 0 si ha un impulso negativo sul pin Borrow che a sua volta è applicato al pin DOWN del contatore CTR2. CTR2 diminuisce di una unità il suo stato mentre CTR1 si porta a valore 9. Nella connessione sincrona **(figura 12 b)**, che fa invece riferimento all'integrato 4510, il clock è applicato contemporaneamente a entrambi i contatori, tuttavia mentre il primo è sempre abilitato (E=1), la stessa cosa non vale per il secondo. Quando il conteggio del primo contatore porta lo stesso al valore finale, il suo pin Carry Out abilità il contatore CTR2. Il successivo impulso di clock determina l'incremento dello strato del secondo contatore e l'azzeramento dello stato del primo. Quanto esposto mette in evidenza le caratteristiche principali dei contatori e la rispettiva logica genera-

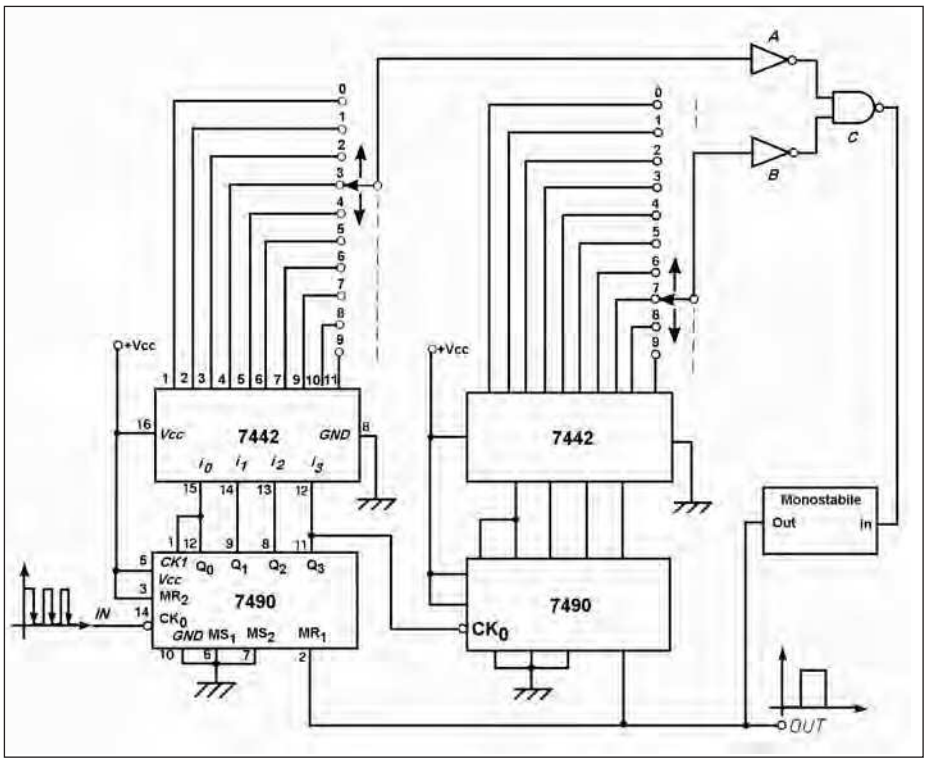


Figura 17: contatore programmabile a due cifre con conteggio 0..99 con impostazione dello stato di fine conteggio (selettore a destra per le decine e a sinistra per le unità) [2].

le di impiego. Spesso, tuttavia essi necessitano di ulteriori circuiti di decodifica per la rappresentazione del conteggio su display a 7 segmenti. Si tratta dei *decoder BCD-display 7 segmenti* come il *7447* sebbene esistano in commercio anche circuiti che integrano entrambe le funzionalità (contatore e decoder) all'interno di uno stesso componente. Un esempio è fornito dalla serie di integrati 74C925..928. Per completezza, oltre all'elenco dei principali circuiti integrati contatori asincroni riportato in **tabella 3**, riportiamo, all'interno della **tabella 4** anche l'elenco di alcuni dei principali contatori sincroni CMOS e TTL più diffusi in commercio.

SIMULAZIONI CIRCUITALI CON CONTATORE 4510 E DECODER PER DISPLAY A 7 SEGMENTI 7447

Il circuito di **figura 13** è analogo a quello riportato nel successivo paragrafo per il contatore TTL7490. La sua implementazione pratica su bread-board o la simulazione attraverso un simulatore circuitale consente di esaminare le funzioni dei diversi pin del contatore CMOS sincrono di tipo BCD 4510. Attraverso il dip-switch (o pulsante) V9 è possibile verificare la funzione di reset del contatore e, attraverso analoghi dip-switch, è possibile condurre prove pratiche sul funzionamento dei comandi di preset e di modalità di conteggio (V8-impostazione della modalità: UP/DOWN). Una batteria di dip-switch può inoltre consentire di impostare a piacimento la parola di preset (Preset Data) che si desidera precaricare all'interno del contatore. Anche il clock può essere simulato attraverso un dip-switch oppure attraverso un pulsante, e la stessa cosa vale per il Clock Enable. Agendo su quest'ultimo è possibile interrompere il conteggio indipendentemente dal clock. Attraverso il comando V10 è inoltre possibile abilitare o disabilitare il decoder. Quando il decoder è disabilitato, il display a sette segmenti appare spento sebbene il conteggio continui a evolvere, ammesso ov-

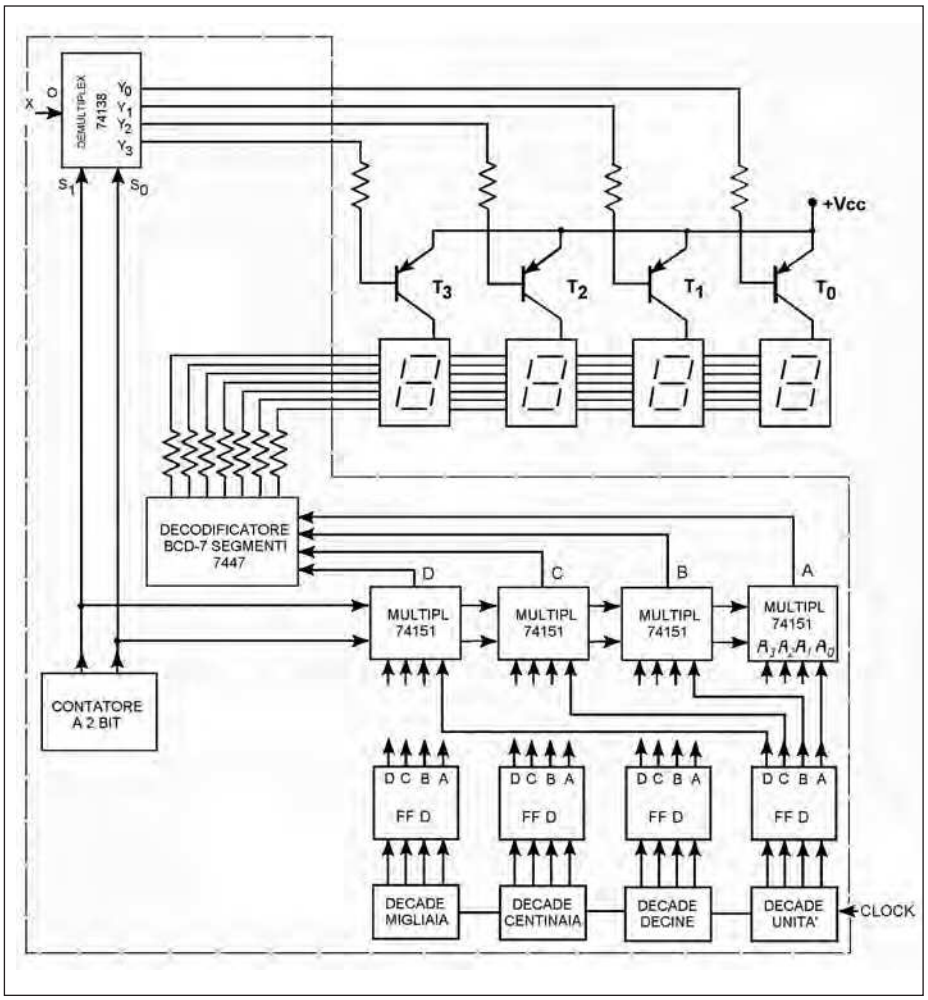


Figura 18: schema di contatore a quattro cifre decimali con display a 7 segmenti multiplexati [2].

viamente che il clock (CP) sia attivo e abilitato (CE=0) e che il comando di preset risulti inattivo (PL=0). Lo stesso circuito può essere espanso in maniera tale da ottenere un contatore a due cifre **(figura 14)**. Tutti i comandi (Preset Enable, Clock Enable, UP/Down e Master Reset) diventano in questo caso comuni ai due contatori a differenza della parola di preset e del clock che è applicato al solo contatore delle unità mentre è propagato da questo al contatore delle decine attraverso il pin TC (Terminal Count). Anche l'abilitazione dei decoder è ovviamente comune alle unità e alle decine. Il funzionamento del circuito è identico pertanto a quello mostrato in **figura 13** a meno del fatto di presentare adesso conteggio tra 0 e 99 invece che tra 0 e 9. Come è ovvio immaginare, lo schema può essere ulteriormente generalizzato, peraltro in maniera ovvia, al fine di aggiungere al conteggio le centinaia, le migliaia e così via.

SIMULAZIONI CIRCUITALI CON CONTATORE 7490 E DECODER PER DISPLAY A 7 SEGMENTI 7447

Lo schema di **figura 15** rappresenta un contatore a una decade implementato impiegando un integrato TTL-7490. La visualizzazione dello stato del contatore è resa possibile da un decoder BCD-7segmenti TTL-7447 e da un display a sette segmenti ad anodo comune. Il funzionamento del circuito può essere ancora una volta simulato attraverso un qualsiasi simulatore di circuiti elettronici oppure sperimentato praticamente su breadboard come effettuato per il contatore 4029. In questo caso, tuttavia, è necessario disporre di qualche componente in più e di una alimentazione a 5 V, per cui non è più possibile utilizzare una pila a 9 V a meno di non prevedere, insieme con essa, anche un opportuno circuito riduttore, per esempio un regolatore di tensione 7805. Con la stessa logica mostrata per il contatore a due cifre decimali implementato attraverso l'integrato 4510, è possibile anche in questo caso espandere il circuito al conteggio di decine, centinaia, migliaia e così via. In **figura 16** è mostrata l'implementazione che consente la realizzazione un conteggio da 0 a 9999 e quindi a quattro cifre decimali. Ancora una volta si tratta di una espansione asincrona in quanto il clock

TABELLA 1: VERSIONI DEL FIRMWARE

SIGLA	TECNOLOGIA	CARATTERISTICHE
74160	TTL	Contatore sincrono analogo al 74163, di tipo BCD, con Clear asincrono
74161	TTL	Contatore sincrono analogo al 74163, con Clear asincrono
74162	TTL	Contatore sincrono analogo al 74163, di tipo BCD
74163	TTL	Contatore sincrono avanti binario a 4 bit con Load e Clear sincroni
74168	TTL	Contatore sincrono avanti-indietro binario a 4 bit con Load sincrono
74169	TTL	Contatore sincrono analogo al 74168, di tipo BCD
74190	TTL	Contatore sincrono avanti-indietro BCD, a 4 bit, Load asincrono, doppia uscita RCO e MAX/MIN per il collegamento in cascata
74191	TTL	Analogo al 74190 ma binario
74192	TTL	Contatore sincrono avanti-indietro BCD, a 4 bit, con doppio ingresso di clock
74193	TTL	Contatore sincrono analogo al 74192 ma di tipo binario
74568	TTL	Contatore sincrono avanti-indietro BCD, a 4 bit, con Load asincrono, Clear sincrono e asincrono ed uscite three-state
74569	TTL	Analogo al 74568 ma di tipo binario
74579	TTL	Contatore sincrono avanti-indietro a 8 bit, con Load sincrono ed uscite three-state; le otto linee di uscita sono bidirezionali
74590	TTL	Contatore sincrono avanti a 8 bit, Clear asincrono, latch in uscita, uscite three-state
74591	TTL	Analogo al 74590 con uscite open-collector
74592	TTL	Contatore sincrono avanti a 8 bit, Load e Clear asincroni, latch in uscita
74593	TTL	Contatore sincrono avanti a 8 bit, Load e Clear asincroni, latch in uscita, uscite three-state; le otto linee di uscita sono bidirezionali
74779	TTL	Contatore sincrono avanti-indietro a 8 bit, Load sincrono, uscite three-state, le uscite sono bidirezionali
74867	TTL	Contatore sincrono avanti-indietro a 8 bit, Load sincrono e Clear asincrono
74869	TTL	Analogo al 74867 ma con clear sincrono
4017	CMOS	Contatore sincrono avanti con clear asincrono con uscite decodificate da 1 a 10
4029	CMOS	Contatore sincrono avanti-indietro a 4 bit, con load asincrono, consente di scegliere tra conteggio binario e BCD
4510	CMOS	Simile al 4516 ma di tipo BCD
4516	CMOS	Contatore sincrono avanti-indietro a 4 bit (binario), presettabile, con Load e Clear asincroni.

Tabella 4: principali contatori sincroni TTL e CMOS.

non è fornito simultaneamente a tutti i contatori ma è propagato da quello delle unità a quelle delle decine e così via fino alle migliaia. L'abilitazione dei decoder delle singole cifre decimali è invece comune ai diversi decoder come il reset dei singoli contatori è comune a tutti e quattro i contatori.

CONTATORI PROGRAMMABILI

Al fine di essere esaustivi sull'argomento, è opportuno accennare ai contatori programmabili. La **figura 17** riporta lo schema di un circuito contatore programmabile a due cifre decimali. Il contatore è in grado di contare da 0 a 99. I commutatori vengono impostati in corrispondenza delle cifre che rappresentano decine e unità, in corrispondenza delle quali si intende bloccare il conteggio (nel caso in figura si tratta del valore 73). I commutatori sono connessi a decodificatori decimali-BCD a quattro bit. Quando il valore di fine conteggio programmato viene raggiunto, attraverso i selettori giungono alle porte inverter A e B due "0", il circuito combinatorio comanda il monostabile il quale genera a sua volta l'impulso di reset che azzera i due contatori.

CONTATORI MULTIPLEXATI

Prima di giungere alle conclusioni intendiamo dare qualche cenno al pilotaggio multiplexato dei display a sette segmenti, tecnica utilizzata sovente non solo nei circuiti contatori ma più in generale in tutti i circuiti digitali che fanno uso di display a segmenti. La necessità nasce da una semplice osservazione: se abbiamo un contatore a n cifre e altrettanti visualizzatori a sette segmenti, ci saranno complessivamente 7xn LED da pilotare. Maggiore è il numero delle cifre e maggiore sarà la corrente assorbita complessivamente. All'aumentare del numero di cifre decimali crescerà inoltre il numero delle resistenze di limitazione che è necessario integrare sulla board e quello dei decoder BCD-7 segmenti che risultano in numero pari al numero di cifre del contatore. Per ovviare a questo dispendio di hardware viene utilizzata la tecnica detta "a display multiplexati" che, sfruttando il fenomeno della persistenza dell'immagine sulla retina dell'occhio umano, consente di ridurre a uno il numero di display accesi in uno stesso istante. In altre parole, in rapida successione viene prima acceso il solo display delle unità, quindi il solo display delle decine e così via. In ciascun istante è pertanto acceso un solo display a sette segmenti per cui il consumo complessivo continua a essere pari a quello di un solo display; inoltre dovendo in ciascun istante essere pilotato un solo display a sette segmenti all'interno del circuito è sufficiente l'impiego di un solo decoder BCD-7 segmenti e di conseguenza anche una sola batteria di resistenze di limitazione. In **figura 18** è riportata la soluzione per un contatore a quattro cifre. Si tratta di una soluzione molto diffusa tanto da essere stata estesa in alcuni casi anche a sistemi in cui queste problematiche si pongono in misura minore, come nel caso dei display LCD. Tutti i segmenti omologhi delle diverse cifre sono collegate a una stessa pista conduttrice. Le sette linee che risultano da questa logica di connessione sono collegate, attraverso altrettante resistenze di limitazione, a un unico decodificatore BCD-7 segmenti. Questo comporta che ciascuna cifra che si presenti al decodificatore viene inviata simultaneamente a tutti i display a sette segmenti. Gli anodi comuni di tutti i display sono collegati all'alimentazione attraverso altrettanti transistor. Questi ultimi sono normalmente interdetti e sono commutati in accensione all'occorrenza attraverso un demultiplexer. Grazie a esso, sebbene sia presente su ciascun display la stessa cifra da visualizzare, questa viene visualizzata sul solo display messo in conduzione dal relativo transistor.

Le cifre del numero da visualizzare sono presentate in successione ai relativi display e l'effetto di persistenza dell'immagine sulla retina fa il resto. Ovviamente, il contatore a due bit deve presentare un clock che consenta alla scansione delle cifre di essere percorsa a una frequenza sufficientemente elevata (generalmente compresa tra 50 e 200 Hz) affinché le cifre rappresentate appaiano all'osservatore come se fossero accese simultaneamente, grazie al fenomeno della persistenza dell'immagine sulla retina, sebbene in realtà esse siano accese in intervalli temporali ben distinti. Ovviamente, lo schema può essere esteso, con le dovute modifiche, a più cifre. In passato questi circuiti erano molto più utilizzati. Nel tempo l'elettronica digitale puramente hardware ha via via lasciato campo all'elettronica programmabile. Oggi l'impiego di circuiti integrati contatori è meno diffusa grazie alla disponibilità di controllori, processori ed elettronica programmabile e l'impiego dei display a 7 segmenti e dei relativi decoder BCD-7 segmenti si è altrettanto ridotta grazie alla possibilità di pilotare direttamente i display LCD attraverso controllori e schede programmabili. Se si fa riferimento per esempio all'impiego di visualizzatori a 7 segmenti, il circuito racchiuso all'interno del bordo continuo di **figura 18** può essere interamente sostituito da un controllore o da un'altra elettronica programmabile.

Se per esempio si analizza la struttura circuitale della scheda di sviluppo Easy-PIC o di altre analoghe della stessa serie Easy ci si accorge che i quattro display a 7 segmenti integrati on-board sono connessi proprio secondo uno schema a multiplexing del tipo riportato in **figura 18**. Tutte le operazioni di gestione del multiplexing sono in questo caso svolte direttamente dal controllore.

CONCLUSIONI

Per ovvi motivi di sintesi e di praticità, si è preferito nel corso dell'esposizione eseguire esperimenti su bread-board solo in relazione all'integrato contatore 4029 mentre per altri si è preferito fare riferimento a semplici esperimenti basati sull'impiego del simulatore circuitale Circuit Maker. Ovviamente, il lettore potrà riprodurre le stesse simulazioni utilizzando anche altri tipi di simulatori circuitali. In ogni caso, quanto esposto dovrebbe risultare sufficiente a comprendere il funzionamento dei contatori. L'auspicio è che il presente articolo possa rappresentare non solo una breve trattazione di carattere pratico sul funzionamento di questi circuiti ma anche e soprattutto una panoramica sintetica e sufficientemente esauriente sull'argomento. Per ulteriori approfondimenti e a complemento di quanto esposto suggeriamo di esaminare i datasheet elencati all'interno del riquadro "Per approfondire". ❑

CODICE MIP 2805103

Una CPU più veloce e con più memoria?
E' arrivato il nuovo CUWIN5000!

CUWIN

CE FC

Se hai bisogno di funzionalità sofisticate prova il CUWIN, un Touch controller rivoluzionario.

Programmabile in C++ o mediante un qualsiasi linguaggio .Net (C#, VB.Net, etc) per la creazione di applicazioni sofisticate che comprendono acquisizione dati, controllo di macchine, monitoraggio di sensori e addirittura reportistica web, solo per citarne alcune.

Più economico di un PC, e con maggiori prestazioni di un controller HMI Touch, prova il CUWIN, un touch controller rivoluzionario già utilizzato in migliaia di applicaizoni in giro per il mondo.

▶ CUWIN5200 ▶ CUWIN5500 ▶ CUWIN6300

- Samsung 2450 (533MHz) ARM Processor
- 128MB SDRAM / 128MB NAND Flash
- Windows® Embedded CE 6.0
- .Net Compact Framework 3.5
- Software Development: C++ or any .Net Language (C#, VB.Net, etc...)
- Ethernet 10/100Base-TX
- RS-232 x 3, RS-485 x 1
- USB 2.0 x 1, USB 1.1 x 1
- Audio (Line Out) / RTC
- Keyboard and Mouse (optional)
- CE, FCC, and KCC certified

CODICE MIP 2814102

elettroshop.com
brilliant electronics since 1998

via Giotto 7
20032 Cormano (MI)
tel. 02 66504794
info@elettroshop.com

www.elettroshop.com

La gestione delle stringhe con mikroBasic

Da questo numero di "Fare Elettronica" inizia un corso avanzato sulla programmazione dei microcontrollori PIC, a grande richiesta dei lettori. Esso può considerarsi a tutti gli effetti il naturale prosieguo del precedente che aveva riscosso, a suo tempo, un successo strepitoso

Le stringhe sono molto importanti nelle applicazioni a microcontrollore e spesso insostituibili. Vediamo come utilizzarle al meglio, adottando anche le funzioni che il mikroBasic mette a disposizione.

LA STRINGA

Come sanno tutti i programmatori, la stringa è una sequenza di caratteri. Se il singolo carattere è formato da 8 bit (byte) la tabella di codifica utilizzata è il codice ASCII, mentre se è formato da 16 bit (word) la codifica è Unicode o UTF. Il codice ASCII mette a disposizione un numero massimo di 2^8=256 caratteri diversi, mentre l'UTF, in via teorica, ne offre 2^16=65536. Nella maggior parte dei casi i microcontrollori utilizzano la prima codifica, dal momento che la memoria utilizzata è piuttosto limitata.
Le stringhe possono contenere spazi, caratteri alfabetici, numerici e segni di punteggiatura, purché facciano parte della codifica utilizzata.
Per delimitare una stringa si usano normalmente i doppi apici. Ad esempio il comando:

```
st = "Salve Mondo"
```

assegna alla variabile *st* (di tipo stringa, dichiarata in modo opportuno) la stringa "Salve Mondo".
La lunghezza di una stringa è semplicemente il numero dei caratteri a essa appartenenti, inclusi spazi e caratteri speciali.

IL CODICE ASCII

Il codice ASCII (American Standard Code for Information Interchange) è un sistema di codifica dei caratteri, a 7 bit oppure a 8 bit, utilizzato nei computer e nei dispositivi elettronici. I programmi lo utilizzano per espletare qualsiasi operazione di input, output e conversione, vedi ad esempio le routine di visualizzazione dei messaggi. I codici decimali da 0 a 31 e il 127 sono caratteri non stampabili, ma servono al sistema per effettuare operazioni di controllo: come il codice 13=CR (Return), 10=LF (Line Feed) e 8=BS (Backspace).

SCOPO DELLE STRINGHE CON I MICROCONTROLLORI

In una applicazione al PC, le stringhe sono molto utilizzate, un po' meno nelle applicazioni a microcontrollore. L'uso prevalente delle stringhe infatti è dedicato alla preparazione e alla gestione dei messaggi di output su unità di visualizzazione LCD. Altre applicazioni possono contemplare l'uso di stringhe di caratteri in una trasmissione codificata, secondo diversi protocolli. Altri utilizzi delle stringhe sono necessari se si vuole, ad esempio, visualizzare un dato numerico: in tal caso è necessaria una conversione in formato stringa per permetterne la gestione corretta dalle funzioni utilizzate.

MICROCONTROLLORE UTILIZZATO

Per tutte le applicazioni del nostro corso, se non diversamente specificato, sarà utilizzato il PIC 18F877. Esso dispone infatti di molta memoria RAM e ROM e nu-

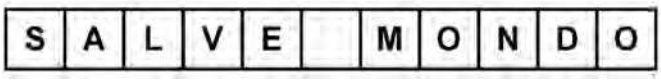


Figura 1: classico esempio di stringa.

merose porte di I/O, nonché di supporti ADC, PWM, RS232 e tanto altro. Inoltre il suo costo è estremamente abbordabile. Gli esempi riportati sono comunque compatibili con la stragrande maggioranza di MCU, dal momento che raramente si effettuerà l'accesso alle risorse dipendenti dall'hardware.

COMPILATORE UTILIZZATO

Il compilatore e l'ambiente di sviluppo utilizzato è il super-collaudato mikroBASIC PRO di MikroElektronika. La sua versione è giunta alla 5.00, almeno alla scrittura del presente articolo. Si consiglia di non utilizzare precedenti versioni in quanto la sintassi di molte funzioni è stata completamente modificata e alcune procedure sono state integralmente riscritte.

DICHIARAZIONE E ASSEGNAZIONE DI UNA STRINGA

Dichiarazione

Come tutte le variabili, anche una stringa deve essere dichiarata prima del suo utilizzo. Una stringa (come nel linguaggio C) è assimilata a un array di caratteri, per cui segue la medesima inizializzazione. La memorizzazione avviene internamente, grazie al codice interno, e la cui strutturazione non interessa tanto al programmatore, che si deve occupare invece di ben altre cose.
Per dichiarare una stringa occorre inoltrare il seguente comando, prima dell'inizio del codice operativo:

```
dim nome_variabile as string
[lunghezza]
```

Ad esempio la dichiarazione seguente predispone una stringa di 20 caratteri:

```
dim titolo as string[20]
```

Il primo carattere è contenuto nell'elemento 0 del vettore, l'ultimo carattere è invece contenuto nell'elemento 20 per permettere la memorizzazione dell'identificatore finale di fine stringa (EOS o carattere ASCII 0). In definitiva dunque sono allocati 21 byte utilizzabili. Non conviene mai sovrascrivere il carattere di fine stringa, per non incorrere in comportamenti imprevisti del compilatore.

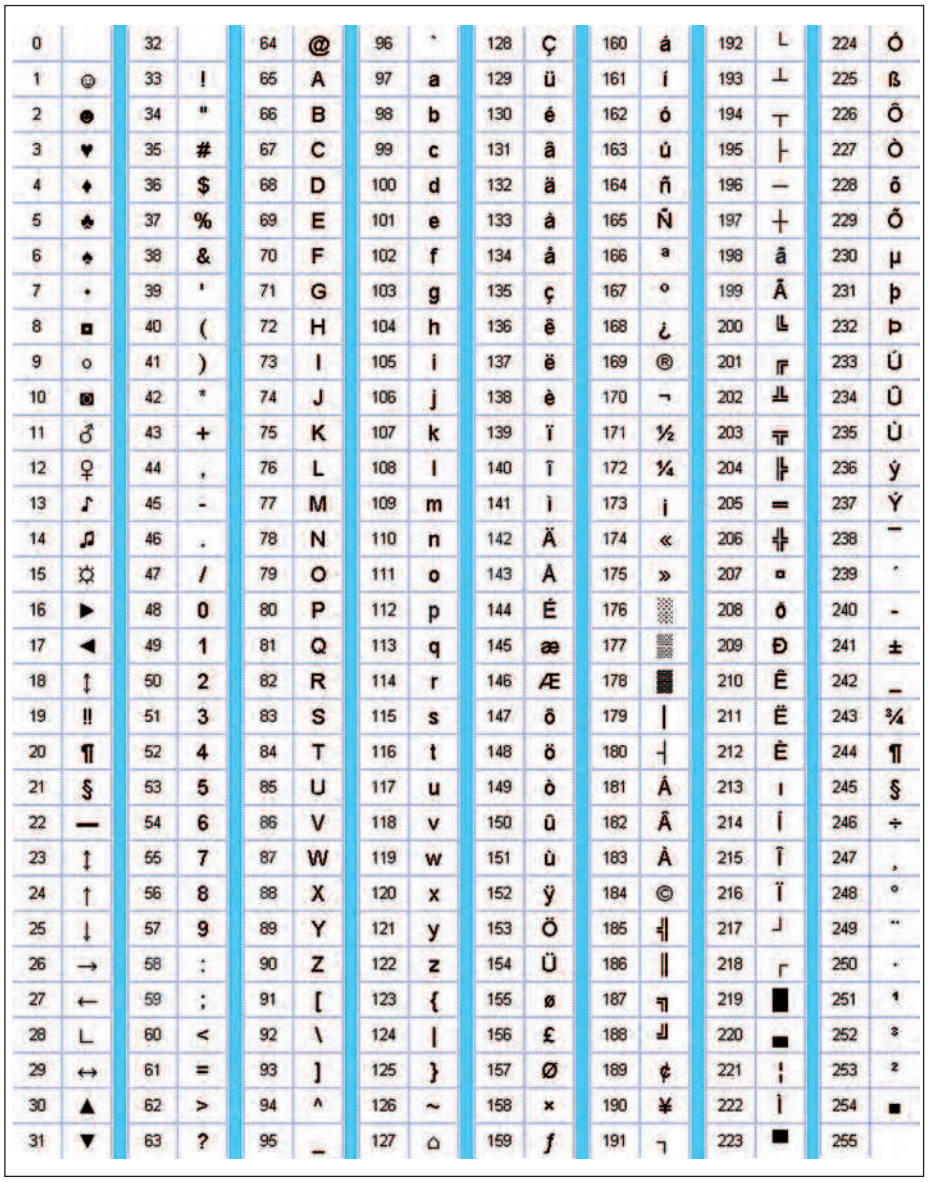


Figura 2: il codice ASCII.

Al momento della dichiarazione, l'intero array di caratteri è riempito con il primo carattere della tabella ASCII (con codice 0), da non confondersi con il carattere 0 (con codice ASCII 48).

Assegnazione

L'assegnazione di un valore di tipo stringa a una stringa avviene come tutte le altri variabili, utilizzando il simbolo "=" per indicare la locazione di destinazione (posta a sinistra dello stesso simbolo) e quella di origine (posta a destra).
Pertanto, utilizzando la variabile prima dichiarata (titolo) è possibile effettuarvi l'assegnazione della frase "Fare Elettronica" scrivendo il seguente programma:

```
program stringa01
dim titolo as string[16]
main:
titolo="Fare Elettronica"
end.
```

Come si vede nella figura 6, l'intero array di caratteri contiene la frase "Fare Elettronica", e ogni carattere risiede in ciascuna locazione di memoria, sequenzialmente. Le posizioni inutilizzate contengono il carattere 0. Il primo carattere del

Figura 3: una stringa con la corrispondente codifica ASCII.

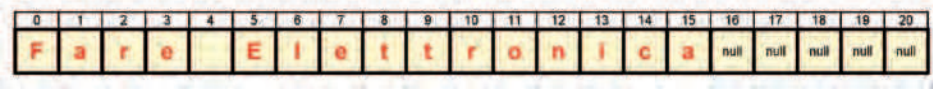


Figura 7: come una stringa memorizza i dati al proprio interno.

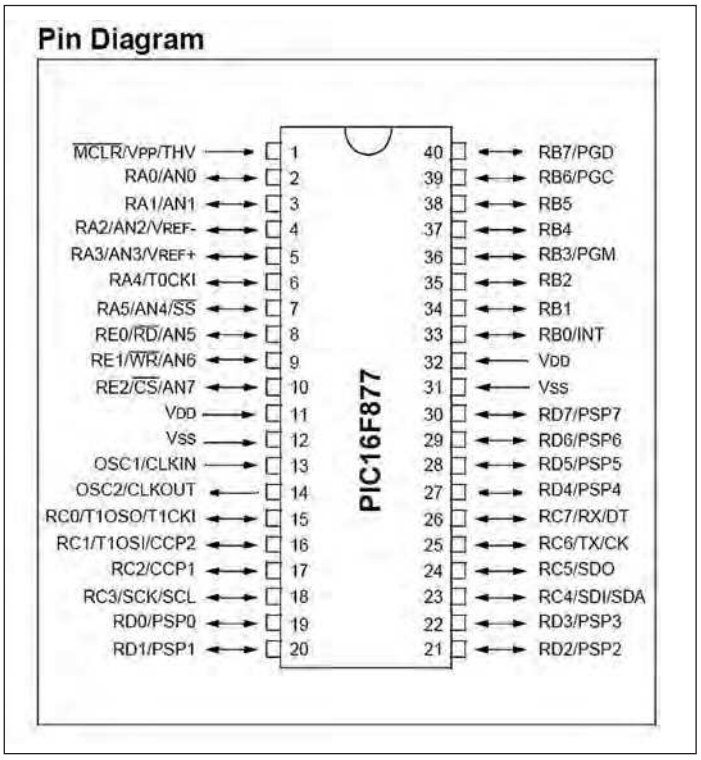


Figura 4: pinout del PIC 16F877.

Name	Value	Address
- titolo	{...}	0x0022
[0]	'□'	0x0022
[1]	'□'	0x0023
[2]	'□'	0x0024
[3]	'□'	0x0025
[4]	'□'	0x0026
[5]	'□'	0x0027
[6]	'□'	0x0028
[7]	'□'	0x0029
[8]	'□'	0x002A
[9]	'□'	0x002B
[10]	'□'	0x002C
[11]	'□'	0x002D
[12]	'□'	0x002E
[13]	'□'	0x002F
[14]	'□'	0x0030
[15]	'□'	0x0031
[16]	'□'	0x0032
[17]	'□'	0x0033
[18]	'□'	0x0034
[19]	'□'	0x0035
[20]	'□'	0x0036

Figura 5: dichiarazione della stringa.

testo è memorizzato nella posizione numero 0 dell'array.
Ogni elemento (e quindi ogni carattere) è direttamente accessibile attraverso il suo indice, ossia il suo numero ordinale, meno uno. Riferendoci dunque all'indice 0 di un vettore, si accede direttamente al suo contenuto, che corrisponde in pratica al codice ASCII del primo carattere in esso memorizzato.

PRIMA ESPERIENZA: CREAZIONE DI UN CHECKSUM

Molto spesso il contenuto della RAM di un microcontrollore non è relativo solo al codice e alle procedure, ma potrebbe riferirsi anche a dati sensibili e di controllo. Si pensi ad esempio al prototipo di un'applicazione che, oltre alle sue tante funzionalità, visualizza su un display LCD il nome dell'azienda produttrice del firmware.
Un abile hacker, o un bravo programmatore di PIC, saprebbe facilmente intercettare e modificare il contenuto della stringa di caratteri contenente il nome aziendale tramite un editor esadecimale. Una buona tecnica per impedire ciò è quella di processare la stringa e sottoporre la stessa a un controllo di *checksum* (somma di controllo). Se la stringa venisse modificata, il programma potrebbe comportarsi di conseguenza bloccando, ad esempio, la sua esecuzione.

Schema elettrico del prototipo

Lo schema elettrico è molto semplice e prevede solo l'adozione del microcontrollore e del display LCD con 16 colonne e 2 righe. Il pin 1 del micro è collegato all'alimentazione attraverso una resistenza di pull-up e ne assicura il funzionamento. Uno stato logico basso ne provoca invece il reset. L'oscillazione di clock è assicurata attraverso i due condensatori ceramici e dal quarzo X1.
Occorre ricordarsi di far corrispondere la frequenza del quarzo con quella del progetto del compilatore, pena lo sfasamento delle temporizzazioni.
Il display LCD è collegato in configurazione "nibble" (4 bit di dati) e, assieme ai segnali di funzionamento, è gestito per intero

Name	Value	Address
- titolo	{...}	0x0022
[0]	'F'	0x0022
[1]	'a'	0x0023
[2]	'r'	0x0024
[3]	'e'	0x0025
[4]	' '	0x0026
[5]	'E'	0x0027
[6]	'l'	0x0028
[7]	'e'	0x0029
[8]	't'	0x002A
[9]	't'	0x002B
[10]	'r'	0x002C
[11]	'o'	0x002D
[12]	'n'	0x002E
[13]	'i'	0x002F
[14]	'c'	0x0030
[15]	'a'	0x0031
[16]	'□'	0x0032
[17]	'□'	0x0033
[18]	'□'	0x0034
[19]	'□'	0x0035
[20]	'□'	0x0036

Figura 6: stringa contenente il testo "Fare Elettronica".

TABELLA 1 FUNZIONI E PROCEDURE DELLA LIBRERIA DI STRINGHE

memchr	**La funzione individua la prima occorrenza di ch byte nelle prime n words in memoria, a partire dall'indirizzo p. La funzione restituisce l'offset di questa occorrenza a partire dall'indirizzo di memoria p o 0xFFFF se ch non è stato trovato.**
memcmp	**La funzione restituisce un valore positivo, negativo o pari a zero indicando la relazione delle prime n words a partire dagli indirizzi p1 e p2. Questa funzione confronta due aree di memoria a partire da p1 e p2, per n words, e restituisce un valore che indica il loro rapporto come segue: <0 p1 "minore" p2 = 0 p1 "uguale a" p2 > 0 p1 "maggiore di" p2**
memcpy	**La funzione copia nn words dall'area di memoria a partire dall'indirizzo di p2 indirizzo alla memoria a partire da p1. Se questi buffer di memoria si sovrappongono, la funzione memcpy non può garantire che le parole vengono copiate con successo. Utilizzare nel caso la funzione memmove.**
memmove	**La funzione copia nn words a partire dall'area di memoria di p2 all'area di memoria a partire da p1. Se questi buffer di memoria si sovrappongono, la funzione memmove assicura che le word di p2 vengono copiati in p1, prima di essere sovrascritte.**
memset	**La funzione riempie i primi n byte nell'area di memoria a partire dall'indirizzo di p con il valore di un carattere.**
strcat	**La funzione aggiunge il valore della stringa s2 alla stringa s1 e termina s1 con il carattere null.**
strcat2	**La procedura confina la stringa s2 alla fine della stringa s1, o al primo carattere null di s1, e mette il risultato nella stringa "I".**
strchr	**La funzione ricerca la stringa s come prima occorrenza del carattere ch. Il carattere di terminazione null s non è incluso nella ricerca. La funzione restituisce la posizione (indice) del primo carattere ch trovato in s. Se nessuna corrispondenza è stata trovata, la funzione restituisce 0xFFFF.**
strcmp	**La funzione confronta lessicograficamente il contenuto delle stringhe s1 e s2 e restituisce un valore che indica il loro rapporto: <0 s1 "minore" s2 = 0 s1 "uguale a" s2 > 0 s1 "maggiore di" s2**
strcpy	**La funzione copia il valore della stringa s2 alla stringa s1 e aggiunge un carattere null alla fine di s1.**
strlen	**La funzione restituisce la lunghezza della stringa s. La lunghezza non include il carattere di terminazione null.**
strncat	**La funzione aggiunge alcuni caratteri della stringa s2 alla stringa s1 e la termina con un carattere null.**
strncpy	**La funzione copia dei caratteri dalla stringa s2 alla stringa s1. Se s2 contiene meno caratteri di "size", s1 è riempito con caratteri nulli fino alla lunghezza complessiva.**
strspn	**La funzione cerca la stringa s1 per i caratteri non trovati nella stringa s2.**
strncmp	**La funzione confronta lessicograficamente i primi n caratteri delle stringhe s1 e s2 e restituisce un valore che indica il loro rapporto: <0 s1 "minore" s2 = 0 s1 "uguale a" s2 > 0 s1 "maggiore di" s2**
strstr	**La funzione individua la prima occorrenza della stringa s2 nella stringa s1 (escluso il carattere di terminazione null).**
strcspn	**La funzione cerca la stringa s1 per uno qualsiasi dei caratteri nella stringa s2.**
strpbrk	**La funzione ricerca s1 per la prima occorrenza di ogni carattere della stringa s2. Il terminatore null non è incluso nella ricerca. La funzione restituisce un indice del carattere corrispondente in s1. Se s1 non contiene caratteri di s2, la funzione restituisce 0xFFFF.**
strrchr	**La funzione ricerca la stringa s per l'ultima occorrenza del carattere ch. Il carattere di terminazione null s non è incluso nella ricerca. La funzione restituisce un indice dell'ultimo carattere trovato in s. Se nessuna corrispondenza è stata trovata, la funzione restituisce 0xFFFF.**
ltrim	**Le procedura elimina gli spazi iniziali della stringa.**
rtrim	**Le procedura elimina gli spazi finali della stringa.**
strappendpre	**La procedura aggiunge caratteri all'inizio della stringa.**
strappendsuf	**La procedura aggiunge caratteri alla fine della stringa.**
length	**La funzione restituisce la lunghezza della stringa passata.**

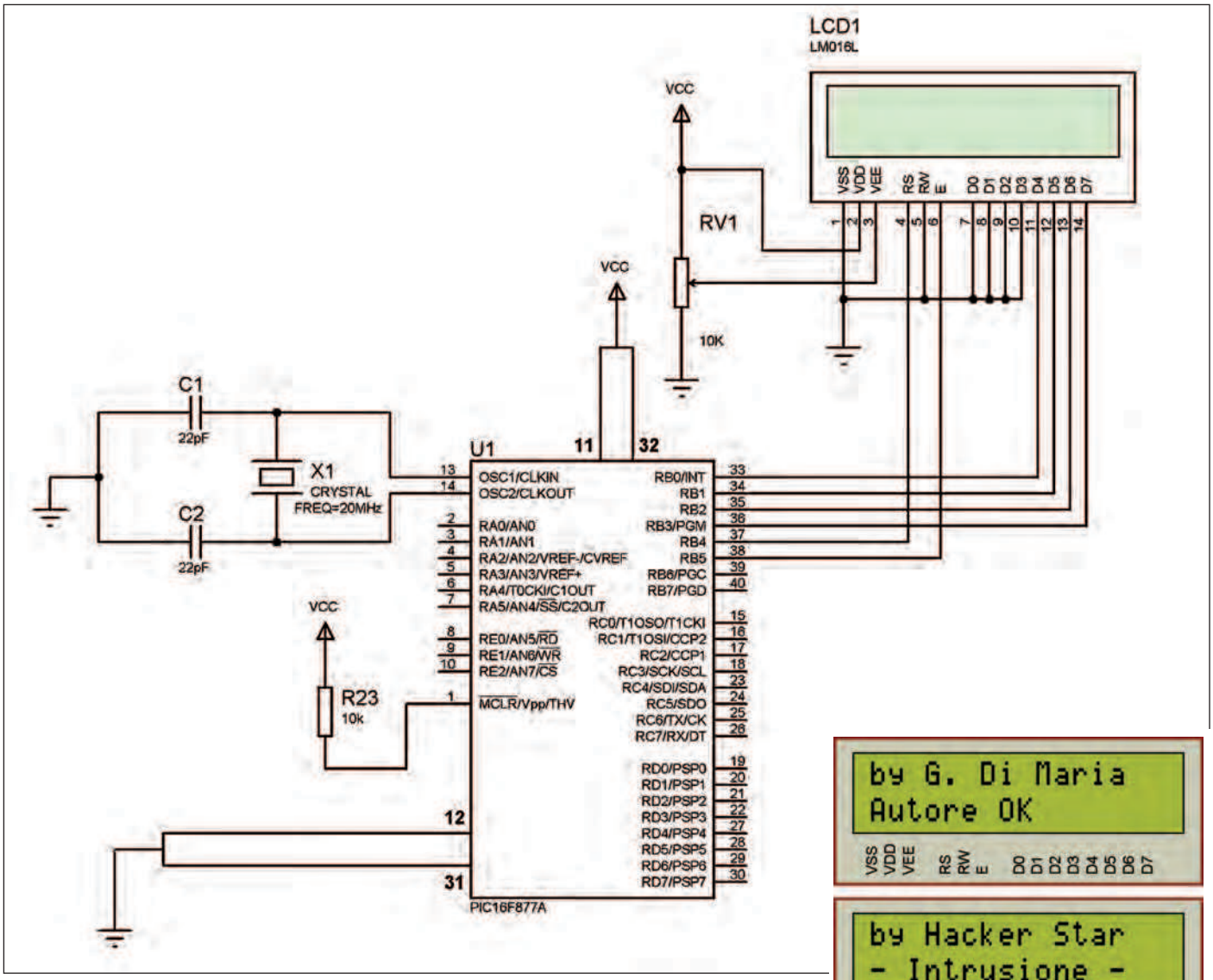


Figura 8: schema elettrico.

Figura 9: i due casi di uscita del controllo del checksum.

dalla Port-B del microcontrollore. Infine il potenziometro RV1 ha la funzione di regolare il contrasto del display, per una visione ottimale.

Il listato

Il **listato 1** perfettamente funzionante è disponibile per il download su www.farelettronica.com. È estremamente semplice. Dopo le dichiarazioni iniziali e il settaggio del display, viene letto dalla memoria il nome dell'autore (Giovanni Di Maria) e memorizzato come stringa nella variabile *autore*.

Quindi, in un ciclo che scandisce tutti i caratteri della stringa, vengono sommati i rispettivi codici ASCII in un accumulatore che funge da sommatore. Dunque, se tale somma equivale a 1095, vuol dire che la stringa è corretta. Se invece un hacker riuscisse a modificare il contenuto della memoria del PIC, la somma finale risulterebbe non compatibile con il controllo e il programma si fossilizzerà in un ciclo infinito di blocco (*halt*), visualizzando anche un messaggio di errore.

Funzionamento e verifica del checksum

Se il programma viene avviato correttamente e non si modifica alcun dato (da parte di pirati informatici), il normale flusso di esecuzione è assicurato. Ma se un hacker riuscisse a modificare le informazioni statiche del PIC (come il nome dell'autore, nell'esempio), il programma si bloccherebbe immediatamente, dal momento che la stringa è ben controllata.

Come si vede dalle immagini, infatti, è sufficiente editare il file .HEX, cercare la zona dati che ospita il nome dell'autore e modificarli di conseguenza. In ogni caso la procedura si bloccherebbe a meno che, con molta fortuna, la somma del nuovo nome impostato fosse nuovamente 1095.

Per prevenire ciò, si può implementare un altro algoritmo che, oltre la somma, computi anche la posizione dei caratteri, in modo da assicurare un protezione molto maggiore.

SECONDA ESPERIENZA: VISUALIZZAZIONE DI NUMERI SUL DISPLAY LCD

Le funzioni e le istruzioni che supportano i display LCD, e provvedono alla visualizzazione dei dati, presuppongono che i loro parametri siano di tipo stringa. In questa maniera è semplicissimo mostrare messaggi, titoli e costanti stringhe sul display. Ma che cosa succederebbe se volessimo visualizzare dei numeri, intesi come variabili o costanti dall'intrinseco valore numerico? Purtroppo questo non può essere effettuato direttamente. Per fortuna il microBasic PRO mette a disposizione alcune funzioni che hanno proprio il compito di trasformare qualsiasi valore numerico in variabile stringa, pronto per essere gestito direttamente dal display. Tali funzioni non restituiscono alcun valore, ma lavorando direttamente sull'indirizzo del parametro stringa (per referenza), costruiscono direttamente l'array di caratteri contenente il valore numerico trasformato.

Ecco di seguito l'elenco di tali funzioni:

- ByteToStr converte un valore *byte* (0 .. 255) in stringa, occupando 3 spazi;
- ShortToStr converte un valore *short* (-127 .. 128) in stringa, occupando 4 spazi;
- WordToStr converte un valore *word* (0 .. 65535) in stringa, occupando 5 spazi;
- IntToStr converte un valore *integer* (-32768 .. 32767) in stringa, occupando 6 spazi;
- LongIntToStr converte un valore *longint* (-2147483648 .. 2147483647) in stringa, occupando 11 spazi;
- LongWordToStr converte un valore *longword* (0 .. 4294967295) in stringa, occupando 10 spazi;
- FloatToStr converte un valore *float* (+/-1.17549435082 * 10^-38 .. +/-6.80564774407 * 10^38) in stringa, occupando 23 spazi;
- WordToStrWithZeros converte un valore *word* (0 .. 65535) in stringa, occupando 5 spazi e riempiendo le posizioni inutilizzate con il carattere '0';
- IntToStrWithZeros converte un valore *integer* (-32768 .. 32767) in stringa, occupando 6 spazi e riempiendo le posizioni inutilizzate con il carattere '0';
- LongWordToStrWithZeros converte un valore *longword* (0 .. 4294967295) in stringa, occupando 10 spazi e riempiendo le posizioni inutilizzate con il carattere '0';
- LongIntToStrWithZeros converte un valore *longint* (-2147483648 .. 2147483647) in stringa, occupando 11 spazi e riempiendo le posizioni inutilizzate con il carattere '0';
- ByteToHex converte un valore *byte* (0 .. 255) in formato esadecimale, occupando 2 spazi;
- ShortToHex converte un valore *short* (-127 .. 128) in formato esadecimale, occupando 2 spazi;
- WordToHex converte un valore *word* (0 .. 65535) in formato esadecimale, occupando 4 spazi;
- IntToHex converte un valore *integer* (-32768 .. 32767) in formato esadecimale, occupando 4 spazi;
- LongWordToHex converte un valore *longword* (0 .. 4294967295) in formato esadecimale, occupando 8 spazi;
- LongIntToHex converte un valore *longint* (-2147483648 .. 2147483647) in in formato esadecimale, occupando 8 spazi;
- StrToInt converte una valore stringa in variabile intera, occupando 6 spazi;
- StrToWord converte una valore stringa in variabile di tipo *word*, occupando 5 spazi.

Le prime sette funzioni trasformano semplicemente i valori numerici in rispettive stringhe. Le successive quattro funzioni effettuano la stessa trasformazione, riempiendo però gli spazi vuoti con '0'. Le ulteriori successive sei funzioni effettuano una conversione da variabile numerico decimale a esadecimale di tipo stringa. Le due ultime funzioni sono molto utili nel caso occorra trasformare un valore stringa in numerico, purché la stringa contenga solamente simboli numerici.

Un piccolo contasecondi

Le funzioni di cui sopra sono insostituibili in tutte le applicazioni nelle quali è necessario visualizzare valori numerici. Anche la realizzazione di un piccolo contasecondi non si sottrae a questa regola. Si tratta di un piccolo programma che visua-

CODICE MIP 2814106

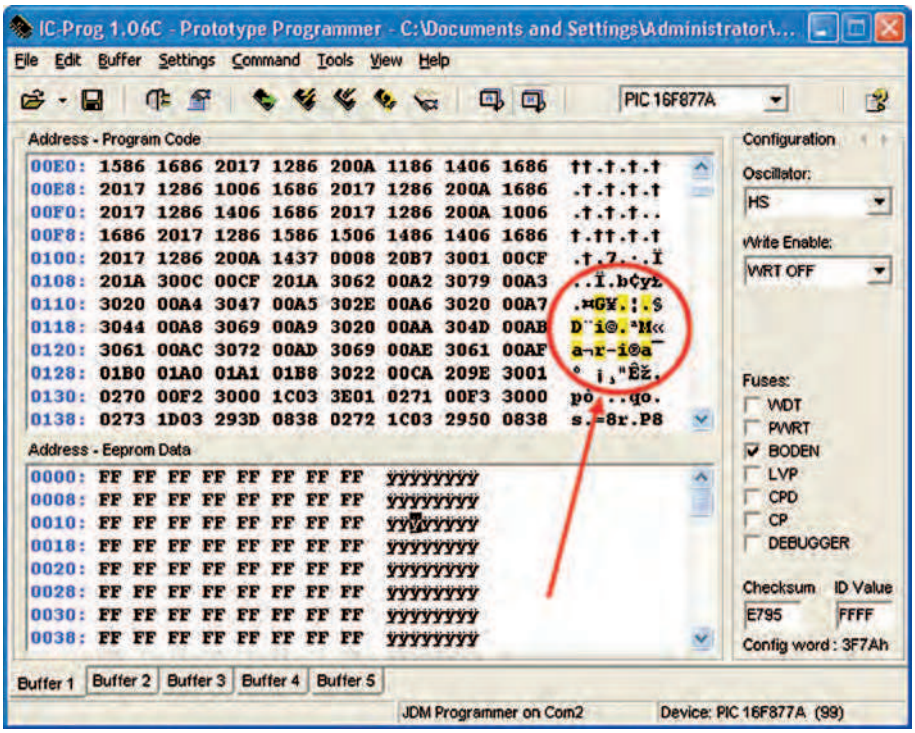


Figura 10: un hacker potrebbe modificare i dati interni del programma.

Figura 11: il display visualizza I numeri, trasformati in stringa.

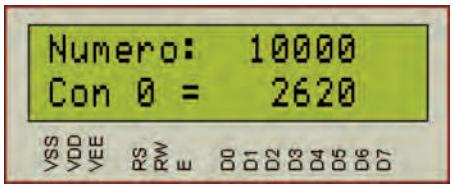


Figura 12: i numeri da 1 a 10000 che contengono almeno uno '0' sono 2620.

lizza, sul display, il conteggio dei secondi trascorsi, da 0 a 59. La conversione del valore numerico dei secondi (variabile *k* di tipo byte) è effettuata grazie alla funzione *ByteToStr* che, qualunque sia il valore del numero (compreso tra 0 e 255) riserva sempre 3 spazi nella stringa. Lo schema elettrico del contasecondi è il medesimo della prima esperienza, mentre il programma relativo è mostrato nel **listato 2** (disponibile su www.farelettronica.com).

TERZA ESPERIENZA: CONTEGGIO DEGLI "ZERI" IN UN NUMERO

Si desidera contare tutti i numeri (da 1 a 10000) che contengano almeno uno zero al loro interno. Su un display si vuol visualizzare, sulla riga superiore, la scansione di tali numeri progressivi, e sulla riga sottostante, il conteggio dei numeri che contengono almeno uno zero. Ad esempio, il numero 102 contiene uno zero, mentre il numero 777 non lo contiene. Per raggiungere lo scopo, invece di utilizzare complicate operazioni matematiche e di statistica, è sufficiente trasformare il contatore in una stringa e, carattere dopo carattere, verificare se al suo interno sia presente il carattere '0'. Nel qual caso si provvede a incrementare un'altra variabile di totalizzazione. Il sorgente, molto semplice, è contenuto nel **listato 3** (disponibile su www.farelettronica.com). Si nota la presenza di due cicli FOR "nidificati". Il primo ciclo, che fa capo alla variabile *k*, processa tutti i numeri interi, da 1 a 10000. Il secondo ciclo, più interno, che fa capo alla variabile *i*, scandisce tutti i caratteri della stringa, alla ricerca di un eventuale zero presente.

LE FUNZIONI STRINGA DI MIKROBASIC

Per conoscere le procedure e le funzioni della libreria di stringhe si osservi la **tabella 1**.

Le funzioni Left, Mid e Right

Allo stato attuale, il compilatore mikroBasic non dispone (stranamente) delle funzioni stringa per ricavare una porzione di testo da sinistra, centro o destra. Poco male, visto che con l'implementazione di una funzione personalizzata, accessibile per riferimento, è possibile realizzare lo scopo. La procedura *str_left* prende un certo numero di caratteri di sinistra, da una stringa d'origine e costruisce quella di destinazione. La procedura *str_mid* prende un certo numero di caratteri, iniziando da una determinata posizione, partendo da una stringa d'origine e costruisce quella di destinazione. Infine La procedura *str_right* prende un certo numero di caratteri di destra, da una stringa d'origine e costruisce quella di destinazione.
Queste tre funzioni sono anche utilizzabili singolarmente e possono essere copiate e incollate nel proprio listato. Esse sono visibili nel **listato 4** (disponibile su www.farelettronica.com).

Figura 13: implementazione delle funzioni Left, Mid e Right.

CONCLUSIONI

Con questo termina la prima puntata del corso sui microcontrollori PIC. Vi diamo appuntamento alle prossime puntate, alla scoperta di ulteriori segreti della programmazione MCU. ❑

CODICE MIP 2805119

Il primo oscilloscopio mixed-signal low-cost!

Oscilloscopio, Analizzatore di spettro

Analizzatore Logico, Generatore Logico

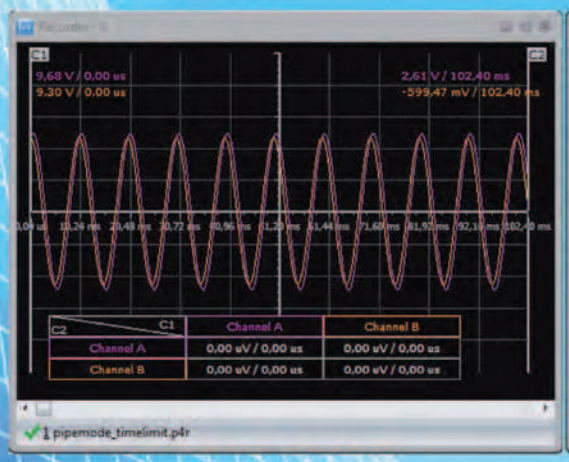
2+5 Canali analogici (12bit, 1MS/s)

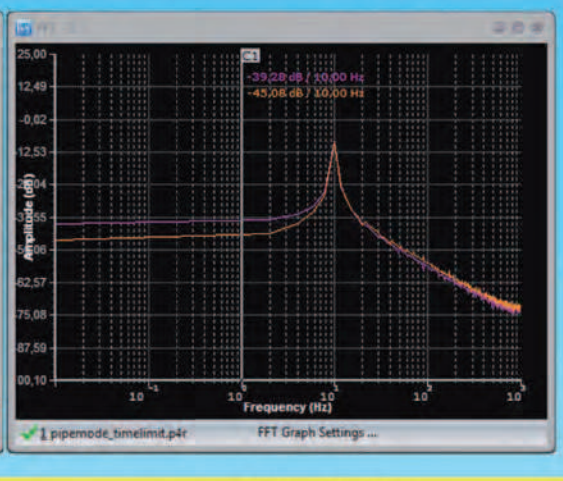
Generatore di onde quadre e PWM

Registratore Digitale

16 Canali Digitali

Aggiornamenti software gratuiti





Oscilloscopio ed analizzatore di spettro

Numero canali: 2+5
Frequenza di campionamento: fino a 1MHz
Memoria:
- Buffer di lettura: 1126 campioni/canale (1 canale), 563 campioni/canale (2 canali).
- Pipe di lettura: 64K campioni/canale (1 o 2 canali).

Massima tensione di ingresso: -20 ÷ +20 V
Risoluzione ADC: 12bits
Triggering:
- Assoluto (per fronti di salita/discesa)
- Differenziale (per differenza tra campioni consecutivi)
- Esterno (per fronti di salita/discesa di segnali TTL)

Funzionalità disponibili: Hamming, Hanning, Blackman, Blackman-Harris.

Analizzatore logico

Numero canali: 16 (8 se utilizzato il generatore logico)
Frequenza di campionamento: fino a 10MHz
Memoria:
- Buffer in lettura (Fs=4-8 MHz) 128 bit/canale.
- Buffer in lettura (Fs=2-2.66 MHz) 1160 bit/canale.
- Buffer in lettura (Fs<=1 MHz) 1544 bit/canale
- Buffer in lettura (in mod. concatenamento) 1 Mbit/canale.
- Pipe di lettura (Fs < 500KHz) 4K a 256 Mbit/canale.

Massima tensione di ingresso: 0 ÷ +5 V
Triggering: per fronti del segnale, maschere, impulsi persi, clock esterno.
Clock: interno/esterno

Registratore

Frequenza di campionamento: fino a 1MHz
Capacità massima di registrazione: 24 ore (Fs < 100 Hz)
Tensione d'ingresso: -20 ÷ +20 V (hardware 2 sub-band)
Risoluzione ADC: 12bits

Generatore logico

Numero canali: 8
Frequenza di campionamento: fino a 1MHz
Memoria: 4000 bit/canale
Tensione di uscita: "0" - 0 V, "1" - 3.3 V
Massima corrente in ingress/uscita: 10 mA

CODICE MIP 2814110

Novità! Il più economico isolatore USB!

PoScope mega1

Ordinali subito su www.ieshop.it/poscope

Corso di Python *(parte sesta)*

INSERIMENTO da *tastiera*

I programmi che abbiamo scritto finora sono piuttosto banali, nel senso che non accettano inserimenti di dati da parte dell'operatore, limitandosi a eseguire sempre le stesse operazioni. Python fornisce un insieme di funzioni predefinite che permettono di inserire dati da tastiera

Tra le funzioni che consentono di inserire dati attraverso la tastiera, la più semplice è raw input. Quando questa funzione è chiamata, il programma si ferma e attende che l'operatore inserisca un comando, confermando poi l'inserimento con Invio (o Enter). A quel punto il programma riprende e raw input ritorna ciò che l'operatore ha inserito sotto forma di stringa:

```
Inserimento=raw_input()
restituisce
Testo inserito
print Inserimento
restituisce
Testo inserito
```

Prima di chiamare raw input è una buona idea stampare un messaggio che avvisa l'operatore di ciò che deve essere inserito. Questo messaggio è chiamato prompt. L'operazione è così comune che il messaggio di prompt può essere passato come argomento a raw input:

```
Nome=raw_input(" Scrivi il tuo nome! ")

restituisce
```

```
Scrivi il tuo nome! Arturo
print Nome
restituisce
Arturo
```

Se il valore da inserire è un intero possiamo usare la funzione input:

```
Prompt=" A che velocità viaggia
il treno? "
```

```
Velocita = input(Prompt)
```

Se l'operatore inserisce una serie di cifre questa è convertita in un intero e assegnata a Velocità. Sfortunatamente se i caratteri inseriti dall'operatore non rappresentano un numero, il programma stampa un messaggio d'errore e si blocca:

```
Velocita = input(Prompt)

A che velocita' viaggia il treno?
ottanta all'ora

SyntaxError: invalid syntax
```

Per evitare questo tipo di errori è generalmente meglio usare la funzione raw input per ottenere una stringa di caratteri e poi usare le funzioni di conversione per ottenere gli altri tipi.

LE FUNZIONI E IL VALORE DI RITORNO

Alcune delle funzioni predefinite che abbiamo usato finora producono risultati: la chiamata della funzione con un particolare argomento genera un nuovo valore che viene in seguito assegnato a una variabile o viene usato come parte di un'espressione.

```
e = math.exp(1.0)

Altezza = Raggio * math.sin
(Angolo)
```

Nessuna delle funzioni che abbiamo scritto sino a questo momento ha ritornato un valore. In questo capitolo scriveremo funzioni che ritornano un valore e che chiamiamo funzioni produttive. Il primo esempio è AreaDelCerchio che ritorna l'area di un cerchio per un dato raggio:

```
import math

def AreaDelCerchio(Raggio):

        temp = math.pi * Raggio**2

        return temp
```

D'altra parte una variabile temporanea come temp spesso rende il programma più leggibile e ne semplifica il debug. Talvolta è necessario prevedere istruzioni di ritorno multiple, ciascuna all'interno di una ramificazione di un'istruzione condizionale:

```
def ValoreAssoluto(x):

        if x ¡ 0:
                return -x
        else:
                return x
```

Questo programma non è corretto in quanto non è prevista un'uscita con return nel caso x sia 0. In questo caso il valore di ritorno è un valore speciale chiamato None:

```
>>>printValoreAssoluto(0)

None
```

LE FUNZIONI BOOLEANE

Le funzioni possono anche ritornare valori booleani (vero o falso) e questo è molto utile per mascherare al loro interno test anche complicati.

```
def Divisibile(x, y):

        if x %y==0:

                return 1

        else:

                return 0
```

Il nome di questa funzione è Divisibile (sarebbe comodo poterla chiamare È Divisibile ma purtroppo gli accenti e le lettere accentate non sono caratteri validi nei nomi di variabili e di funzioni).
E consuetudine assegnare nomi che sembrano domande con risposta sì/no alle funzioni booleane: Divisibile? Bisestile? NumeroPari? Nel nostro caso Divisibile ritorna 1 o 0 per indicare se x è divisibile o meno per y. Vale il discorso già fatto in precedenza: 0 indica falso, qualsiasi valore diverso da 0 indica vero.
Possiamo rendere le funzioni ancora più concise avvantaggiandoci del fatto che la

condizione nell'istruzione if è anch'essa di tipo booleano:

```
def Divisibile(x, y):

        if x %y==0:
```

Questa sessione mostra la nuova funzione in azione:

```
def Divisibile(6,4):
        0
def Divisibile(6,3):

        1
```

LA RICORSIONE

Finora abbiamo imparato una piccola parte di Python, ma potrebbe essere utile sapere che questo sottoinsieme è già di per sé un linguaggio di programmazione completo: questo significa che con gli elementi che già abbiamo a disposizione è possibile esprimere qualsiasi tipo di elaborazione. Aggiungendo solo qualche comando di controllo per gestire tastiera, mouse, dischi ecc. qualsiasi programma potrebbe già essere riscritto usando solo le caratteristiche del linguaggio che conosciamo finora. La prova di questa affermazione è un esercizio non banale e fu dimostrata per la prima volta da Alan Turing, uno dei primi teorici dell'informatica (in realtà era un matematico, ma molti degli informatici di allora erano matematici). Di conseguenza la dimostrazione viene chiamata Teorema di Turing. Per dare un'idea di che cosa si può fare con ciò che sappiamo finora, proveremo a valutare alcune funzioni matematiche definite ricorsivamente. Una funzione ricorsiva è simile a una definizione circolare, nel senso che la sua definizione contiene un riferimento a ciò che viene definito.

D'altra parte osservando la definizione della funzione matematica fattoriale (indicata da un numero seguito da un punto esclamativo) ci si accorge che la somiglianza è notevole:

```
n!=1

n(n-1)!
```

Questa definizione stabilisce che il fattoriale di 0 è 1 e che il fattoriale di ogni altro valore n è n moltiplicato per il fattoriale di n–1.

Così 3! è 3 moltiplicato 2!, che a sua volta è 2 moltiplicato 1!, che a sua volta ancora è 1 moltiplicato 0!, che per definizione è 1. Mettendo tutto assieme: 3! è uguale a 3 per 2 per 1, e cio è pari a 6. Se si scrive una definizione ricorsiva, solitamente si può anche scrivere un programma Python per valutarla. Il primo passo è quello di decidere quali siano i parametri da passare alla funzione.

```
Fattoriale ha un solo parametro:
def Fattoriale(n):

        Se l'argomento è 0 dobbiamo ritornare il valore 1:

def Fattoriale(n):

        if n == 0:

        return 1
```

Altrimenti, e questa è la parte interessante, dobbiamo fare una chiamata ricorsiva per trovare il fattoriale di n – 1 e poi moltiplicare questo valore per n:

```
def Fattoriale(n):

if n == 0:

        return 1

else:

        FattorialeMenoUno = Fattoriale(n-1)

        Risultato = n * FattorialeMenoUno

        return Risultato
```

Il flusso di esecuzione del programma è simile a quello di ContoAllaRovescia precedentemente discusso. Se chiamiamo Fattoriale con il valore 3: Dato che 3 non è 0, seguiamo il ramo else e calcoliamo il fattoriale di n=3–1=2... Dato che 2 non è 0, seguiamo il ramo else e calcoliamo il fattoriale di n=2–1=1... Dato che 1 non è 0, seguiamo il ramo else e calcoliamo il fattoriale di n=1-1=0... Dato che 0 è 0 ritorniamo 1 senza effettuare ulteriori chiamate ricorsive. Il valore di ritorno (1) è moltiplicato per n (1) e il risultato (1) restituito alla funzione chiamante. Il valore di ritorno (1) è moltiplicato per n (2) e il risultato (2) restituito alla funzione chiamante. Il valore di ritorno (2) è moltiplicato per n (3) e il risultato (6) diventa il valore di ritorno della funzione che ha fatto partire l'intero processo. Nota come nell'ultimo frame le variabili locali FattorialeMenoUno e Risultato non esistono perchè il ramo che le crea non viene eseguito.

CONSIDERAZIONI

Seguire il flusso di esecuzione è un modo di leggere i programmi, ma può dimostrarsi piuttosto difficile da seguire man mano che le dimensioni del codice aumentano. Un modo alternativo è ciò che potremmo chiamare accettazione con fiducia: quando si arriva a una chiamata di funzione invece di seguire il flusso di esecuzione si parte dal presupposto che la funzione chiamata si comporti correttamente e che ritorni il valore che ci si attende. In ogni modo si sta già praticando questa accettazione con fiducia quando si usano le funzioni predefinite: quando si chiama math.cos o math.exp non si va a controllare l'implementazione delle funzioni, assumendo che chi le ha scritte fosse un buon programmatore e che le funzioni siano corrette. ❑

CODICE MIP 2805091

CODICE MIP 2813674

CERCASI

Affermata Società italiana nel settore telecomunicazioni broadcast seleziona progettisti esperti in elettronica digitale, microprocessori, FPGA, protocolli, sviluppo firmware e software.

Inviare curriculum a:
prog2008@libero.it

Next Generation Intelligent LCDs

RISPARMIA TEMPO.
RISPARMIA DENARO.
RISPARMIA MANODOPERA.

Riduci significativamente i costi di sviluppo

Time-to-market ultra-veloce per le tue applicazioni

Per maggiori informazioni visita il sito **www.elettroshop.com/demmel** oppure chiama il numero **02/66504794**

Corso SystemVerilog *(parte sesta)*

INTRODUZIONE *alla* libreria OVM

Concetti di base, interfacce, transazioni e driver

Nella precedente puntata abbiamo visto in generale come si compone l'architettura di un test bench di verifica funzionale, discutendo caratteristiche e metodi dei diversi componenti, come driver, agent, sequencer, environment e test. Sfruttando le caratteristiche del linguaggio SystemVerilog abbiamo inoltre sviluppato dei modelli di alcuni di questi componenti. Tali modelli sono apparsi davvero semplici e mancano di molte delle funzionalità di configurabilità e controllo che sono necessarie per ogni tipica applicazione. Per fortuna, modelli completi in questo senso sono oggi disponibili all'interno della libreria OVM unitamente a un ambiente strutturato e riconfigurabile. La libreria è open-source e può essere scaricata gratuitamente a partire dalla pagina web indicata nei riferimenti al punto [1], aprendo il link "Kit Downloads & User Contributions"; tuttavia, a oggi, non esistono ancora simulatori SystemVerilog gratuiti che la supportino completamente. VerificationAcademy è il portale dedicato alla libreria OVM e alla più recente metodologia UVM. Correntemente mantenuto da Mentor Graphics, rende disponibili, previa registrazioni gratuita, una serie di interessanti tutorial su aspetti di base e avanzati, oltre a includere un forum per le discussioni e un cookbook costantemente aggiornato rispetto alle ultimi revisioni della libreria. Lungi dal voler coprire tutti gli aspetti, nelle prossime puntate del nostro corso sul linguaggio SystemVerilog, vedremo le caratteristiche principali della libreria OVM e dei componenti inclusi attraverso la discussione dell'esempio di riferimento che viene distribuito unitamente a essa. Seguiremo un approccio bottom-up partendo dalla descrizione del driver per arrivare a quella della sequenza di test da simulare.

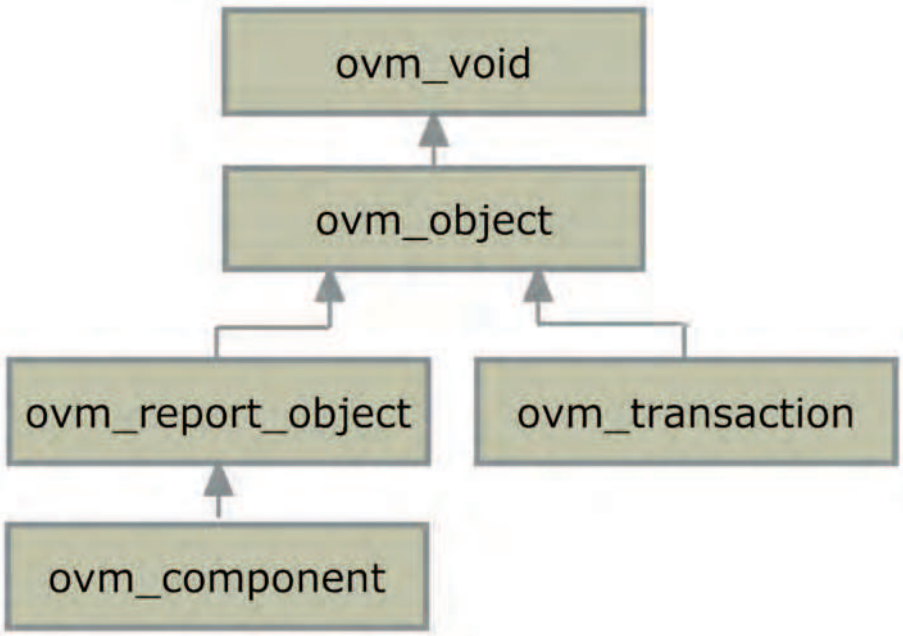


Figura 1: l'organizzazione delle classi base (da [3]).

LA STRUTTURA DELLA LIBRERIA OVM

Prima di vedere in dettaglio come realizzare mediante la libreria OVM i diversi componenti di un test bench per la verifica funzionale, può essere utile dare uno sguardo di massima a come la libreria stessa è strutturata. Essa è organizzata in un insieme di classi che comprende, in linea di principio, i seguenti gruppi:

- *base*, che include, come mostrato schematicamente in **figura 1**, gli oggetti fondamentali come componenti generici, transazioni e porte (queste ultime intese come implementazioni delle interfacce di comunicazione tra componenti);

• *reporting*, che definisce gli oggetti con i metodi per la creazione di log e report di simulazione;
• *factory*, che implementa oggetti e metodi per la gestione, appunto, di una *factory*, intendendo con tale termine, in accordo con l'accezione presente anche in altri linguaggi di programmazione a oggetti, una struttura dati della quale esiste una sola istanza (*singleton*) e che serve come costruttore polimorfico per la registrazione di oggetti e l'override dei relativi metodi e proprietà;
• *synchronization*, che implementa metodi e oggetti per la sincronizzazione dei processi;
• *policies*, che definisce metodi standard per le funzioni di printing, comparing, recording, packing e unpacking di oggetti derivati dalla classe ovm_object;
• *tlm*, che include implementazioni astratte delle interfacce di comunicazione transaction-level (del tipo introdotto in una delle precedenti puntate del corso) e classi generiche (essenzialmente *port* ed *export*) per facilitarne l'uso;
• *components*, che definisce classi, metodi e proprietà per i componenti tipici di un test bench come driver o agent.

COMPONENTI

Nella **figura 2**, in particolare, è mostrato proprio l'organizzazione delle classi presenti nell'insieme di *components* della libreria OVM. Di seguito ci riferiremo quasi sempre a tali classi con il termine generico di componenti, volendo intendere con questo classi che ereditano da ovm_component.
Il contesto in cui il termine componente sarà usato di volta in volta dovrebbe essere sufficiente a chiarire di per sé se s'intende tale l'accezione oppure quella più generale (non riferita all'ambiente OVM) di costituente (come, ad esempio, modulo o entità).
In un certo qual modo, i componenti OVM sono del tutto analoghi ai moduli in linguaggio Verilog; possono infatti essere create, ad esempio, gerarchie di componenti con oggetti che ne istanziano al loro interno altri connessi tra loro. Tuttavia essendo classi e non oggetti statici, i componenti OVM sono creati a run time e non durante la fase di elaborazione gestita dal simulatore SystemVerilog. La creazione e connessione dei componenti nelle gerarchie previste diventa così responsabilità della libreria OVM stessa che deve fornire metodi specifici.
La procedura prevista si basa su di un insieme di fasi eseguite in sequenza. Ogni fase è rappresentata da un metodo virtuale (task o funzione a seconda che gli sia consentito o meno di consumare tempo di simulazione) con un'implementazione triviale nella classe base ovm_component, ridefinibile dagli oggetti derivati. Un controller specifico definito dalla libreria OVM (ma invisibile all'utente) gestisce le transizioni di fase, terminando quella corrente e procedendo alla successiva dopo aver eseguito i metodi relativi per tutti i componenti presenti nella gerarchia creata.
Le principali fasi previste, elencate nell'ordine nel quale sono eseguite all'interno della libreria, sono le seguenti:
• *build*, durante la quale sono instanziati e configurati componenti, port ed export. Tali oggetti sono stati precedentemente creati mediante chiamata dei relativi costruttori new();
• *connect*, durante la quale sono realizzate le connessioni tra i componenti costruiti in precedenza, secondo la gerarchia specificata dall'utente;
• *end_of_elaboration*, durante la quale è possibile applicare modifiche alla configurazione elaborata;
• *run*, durante la quale sono eseguite le funzionalità specifiche di ogni componente che ne dettano il comportamento; si tratta dell'unico metodo predefinito realizzato mediante task. I metodi run() di tutti i componenti instanziati e configurati sono eseguiti in parallelo;
• *extract*, durante la quale sono collezionate informazioni di vario tipo (ad esempio, di coverage) dei componenti configurati;
• *check*, durante la quale è verificata la correttezza dei dati di validazione e di controllo estratti nella precedente fase;
• *report*, durante la quale sono creati log e report sulla base della informazioni raccolte e verificate.

LISTATO **1**

```
interface xbus_if;

  // Control flags
  bit                has_checks = 1;
  bit                has_coverage = 1;

  // Actual Signals
  logic              sig_clock;
  logic              sig_reset;
  logic       [15:0] sig_request;
  logic       [15:0] sig_grant;
  logic       [15:0] sig_addr;
  logic        [1:0] sig_size;
  logic              sig_read;
  logic              sig_write;
  logic              sig_start;
  logic              sig_bip;
  wire logic   [7:0] sig_data;
  logic        [7:0] sig_data_out;
  logic              sig_wait;
  logic              sig_error;

  logic              rw;

assign sig_data = rw ? sig_data_out : 8'bz;

endinterface : xbus_if
```

IL PROTOCOLLO XBUS

L'esempio incluso nella distribuzione standard della libreria OVM implementa un ambiente di verifica funzionale per un bus denominato XBUS. Il protocollo è non standard ed è adottato principalmente per scopo didattico. Le caratteristiche principali sono:

- protocollo non-multiplexed, sincrono, non-pipelined;
- fasi di indirizzamento e trasferimento dati sequenziali;
- bus indirizzi a 16 bit;
- bus dati ad 8 bit (per evitare problemi di allineamento);
- supporto multi-master e arbitraggio mediante semplice schema request/grant;
- supporto per trasferimenti a burst;
- supporto per inserzioni di wait-state da parte del dispositivo slave indirizzato.

Per i dettagli del protocollo si rimanda alla descrizione presente all'interno della guida utente [2] della libreria OVM. In questa sede, e per quanto concerne la trattazione che segue, è sufficiente rimandare alla **figure 3a** e **3b** che mostrano schematicamente accessi in lettura e in scrittura sul bus. Nei prossimi paragrafi vedremo passo passo come l'ambiente di verifica funzionale per tale bus è creato nell'esempio incluso nella libreria OVM.

INTERFACCIA E TRANSAZIONI

Il primo passo nella costruzione di un ambiente di verifica funzionale per bus XBUS è la definizione dell'interfaccia (nel senso che questo termine assume in SystemVerilog, come discusso nella precedente puntata del nostro corso) che racchiude i segnali del bus e della classe. Il **listato 1** riporta un estratto del codice relativo con la definizione dell'insieme di segnali previsti dal protocollo. Se si guarda al codice completo incluso nella distribuzione della libreria, oltre a ciò, si troveranno definite anche alcuni funzioni di verifica di correttezza di protocollo, come ad esempio il controllo su eventuali valori indefiniti dell'indirizzo durante la fase di indirizzamento, sulla corretta durata del segnale di reset o sulla non contemporanea asserzione delle flag di read e write. Una generica transazione sul bus viene invece descritta mediante un oggetto derivato dalla classe ovm_sequence_item che a sua volta eredita da ovm_transaction **(figura 1)**. Il **listato 2** ne riporta anche in questo caso un estratto del codice di esempio. Oltre alla definizione delle caratteristiche distintive di una specifica transazione, come l'indirizzo *addr* cui accedere, il buffer dati *data[ ]* da scrivere o letto (realizzato come array associativo), il tipo di accesso *read_write* (in lettura o scrittura) ed il numero di wait state inseriti dallo slave nel trasferimento, sono presenti anche (righe 12-24) costrutti specifici che definiscono le costrizioni da applicare per la randomizzazione dei diversi parametri. Non abbiamo avuto modo di parlare di questo aspetto del linguaggio in precedenza. Per quello che può servire al momento, è sufficiente sottolineare come il linguaggio SystemVerilog consenta di dichiarare una proprietà di una classe di tipo rand e quindi definisca

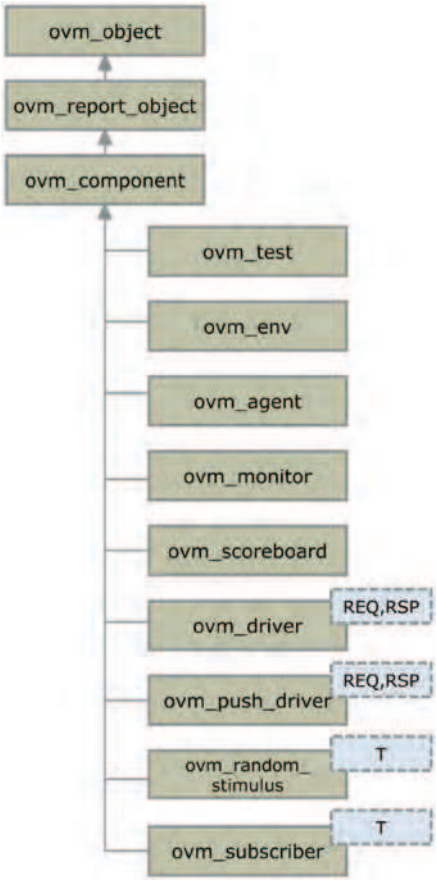


Figura 2: l'organizzazione delle classi di tipo components (da [3]).

LISTATO **2**

```
 class xbus_transfer extends ovm_sequence_item;
  rand bit [15:0]           addr;
  rand xbus_read_write_enum read_write;
  rand int unsigned         size;
  rand bit [7:0]            data[];
  rand bit [3:0]            wait_state[];
  rand int unsigned         error_pos;
  rand int unsigned         transmit_delay = 0;
  string                    master = "";
  string                    slave = "";

  constraint c_read_write {
    read_write inside { READ, WRITE };
  }
  constraint c_size {
    size inside {1,2,4,8};
  }
  constraint c_data_wait_size {
    data.size() == size;
    wait_state.size() == size;
  }
  constraint c_transmit_delay {
    transmit_delay <= 10 ;
  }
endclass : xbus_transfer
```

un metodo per la classe stessa denominato randomize() che consente di assegnare valori casuali alla suddetta proprietà. È possibile imporre vincoli al processo di randomizzazione, come appunto specificato nel **listato 2**, nel qual caso, ad esempio, il numero di byte del trasferimento corrente può essere scelto casualmente tra i valori 1, 2 4, o 8.

DRIVER

ovm_driver è la classe base fornita dalla libreria OVM per la modellizzazione di driver. Il **listato 3** ne riporta la relativa dichiarazione; REQ e RSP rappresentano rispettivamente generiche transazioni in richiesta e risposta.
Il componente dispone delle porte:

• seq_item_port, per la richiesta della transazioni da eseguire al sequencer (nelle prossime puntate vedremo come tale comunicazione viene espletata);
• rsp_port, per l'inoltro dell'eventuale relativa risposta.
Fornisce poi il metodo *new(string name, ovm_component parent)* per creare una istanza della suddetta classe; il campo *name* rappresenta il nome dell'istanza creata mentre *parent* l'handle del componente al livello superiore nella gerarchia (ove esistente).
In generale, la procedura da seguire per create un driver specifico per la propria applicazione consiste quindi nel derivare, appunto, una nuova classe da ovm_driver. Tale nuova classe potrà essere dotata (ove necessario) di metodi base per le funzioni di copia, stampa e confronto sulla base delle macro fornite dalla libreria stessa e dovrà includere una descrizione, almeno, del task per la fase di run(). In linea di principio, tale task sarà strutturato così da includere chiamate:
• alla funzione get_next_item() della porta seq_item_port per la richiesta della transazione da eseguire;
• a metodi propri per consumare la transazione restituita dal sequencer, attivando come specificato i segnali di interfaccia;
• alla funzione item_done della porta seq_item_port per segnale al sequencer che ha originato la transazione l'esecuzione di questa.
Deve inoltre essere istanziata un'interfaccia virtuale per la connessione del driver al DUT nel test bench e un metodo per la sua associazione a livello più alto. Nella puntata precedente abbiamo già chiarito la necessità di utilizzare per questo un'interfaccia virtuale, come risultato della modalità adottata dal linguaggio SystemVerilog di passare per valore i parametri nelle chiamate a task e funzioni.
Il **listato 4** riporta un estratto del codice per la descrizione del driver di una periferica master per bus XBUS, come nell'esempio di riferimento incluso nella distribuzione della libreria OVM.
Si notino, in particolare, l'instanza dell'interfaccia virtuale (riga 4) e la descrizione del costruttore new() (righe 15-17) che richiama il costruttore della classe ovm_driver. Super è una parola chiave

LISTATO 3

```
class ovm_driver #(
  type REQ = ovm_sequence_item,
  type RSP = REQ
) extends ovm_component
```

LISTATO 4

```
class xbus_master_driver extends ovm_driver #
(xbus_transfer);

  // The virtual interface used to drive and view HDL
  // signals.
  protected virtual xbus_if xmi;

  // Master Id
  protected int master_id;

  // Provide implmentations of virtual methods such
  // as get_type_name and create
  `ovm_component_utils_begin(xbus_master_driver)
    `ovm_field_int(master_id, OVM_ALL_ON)
  `ovm_component_utils_end

  // new - constructor
  function new (string name, ovm_component parent);
    super.new(name, parent);
  endfunction : new

  // assign_vi
  function void assign_vi(virtual interface xbus_if xmi);
    this.xmi = xmi;
  endfunction : assign_vi

  // run phase
  virtual task run();
    fork
      get_and_drive();
      reset_signals();
    join
  endtask : run
…
endclass : xbus_master_driver
```

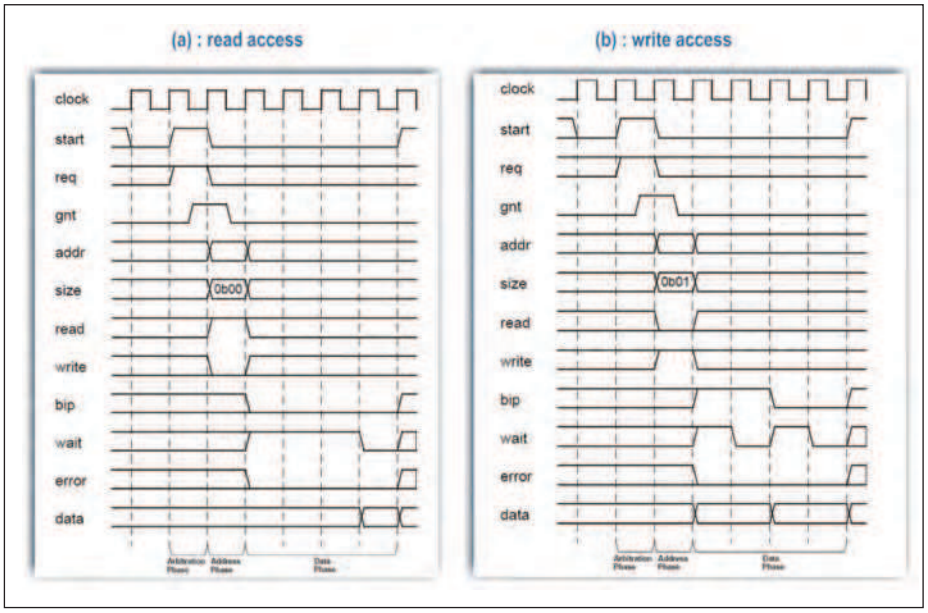


Figura 3: protocollo XBUS : accessi in lettura (a) e scrittura (b) (da [2]).

in SystemVerilog ed è utilizzata nelle classi derivate per riferirsi a metodi della classe parente; è necessario utilizzarla, ad esempio, per riferirsi a metodi della classe parente che siano sovrascritti dalla classe derivata. La dichiarazione della classe viene parametrizzata (riga 1) rispetto alla transazione particolare del bus XBUS. La funzione assign_vi() (righe 20-22) consente di assegnare un'interfaccia specifica all'interfaccia virtuale locale alla classe mentre le macro 'ovm_component_utils e successive (righe 10-12) consentono di definire per la classe corrente implementazioni di base di alcuni metodi standard, come create(), oltre a registrare il driver stesso all'interno della factory. Il task run, infine, (righe 25-30) esegue in parallelo i metodi locali get_and_drive() e reset_signals(). Il **listato 5** mostra in particolare l'implementazione del primo. Seguendo lo schema cui si è accennato in precedenza, il metodo dapprima attende che il segnale di reset sia rilasciato (riga 35), quindi entra in un ciclo infinito (righe 36-43). Per ogni iterazione, richiede una transazione da eseguire al sequencer; il metodo gext_next_item() è di tipo blocking e quindi non ritorna fino a che non ve ne è una disponibile. Ottenuta una transazione, crea la risposta (riga 39) clonando la richiesta generata dal sequencer; pilota infine i segnali dell'interfaccia in accordo a quest'ultima (riga 41) e segnala al sequencer (riga 42) il termine dell'esecuzione della richiesta ricevuta. Per i dettagli del metodo drive_transfer() si rimanda al codice completo disponibile nell'esempio di riferimento della libreria. Nel leggere il codice fare riferimento alle figure 3a e 3b riportate in precedenza e che mostrano i dettagli del protocollo di accesso in lettura e scrittura.

LISTATO **5**

```
// get_and_drive
  virtual protected task get_and_drive();
    @(negedge xmi.sig_reset);
    forever begin
      @(posedge xmi.sig_clock);
      seq_item_port.get_next_item(req);
      $cast(rsp, req.clone());
      rsp.set_id_info(req);
      drive_transfer(rsp);
      seq_item_port.item_done(rsp);
    end
  endtask : get_and_drive
```

CONCLUSIONI

In questa puntata abbiamo introdotto la struttura della libreria OVM e descritto la realizzazione mediante questa dei primi tre elementi base di un ambiente di verifica funzionale, ovvero: l'interfaccia per i segnali del protocollo, la classe per la descrizione delle transazioni supportate dallo stesso e il driver. Tali elementi coprono in qualche modo quello che potrebbe essere considerato come il livello di segnale del protocollo. Il driver traduce, infatti, come abbiamo altre volte accennato, le transizioni dei segnali dell'interfaccia in transazioni. Nella prossima puntata vedremo come descrivere l'infrastruttura per la gestione di tali transazioni a più alto livello mediante creazioni di sequencer e agent. ❑

CODICE MIP 2805087

PER *approfondire...*

[1] www.verificationacademy.com

[2] OVM User Guide - Version 2.1.2, June 2011

[3] OVM Class Reference - Version 2.1.2, June 2011

[4] Glasser, M. "Open Verification Methodology Cookbook" – Springer, 2009

Un'offerta formativa unica nel suo genere: corsi sulle tematiche più attuali e sull'apprendimento delle tecniche di progettazione aggiornate

CORSI A CATALOGO 2012
WWW.FORTRONIC.IT

TEMATICHE

LED & DISPLAY
- LED TECNOLOGY
- FONDAMENTI DI LED LIGHTING DESIGN
- CERTIFICAZIONI e NORMATIVE

RF & WIRELESS
- INTRODUZIONE ALLE COMUNICAZIONI SHORT RANGE
- IL PROTOCOLLO DI ZIGBEE

PROGETTAZIONE
- PROGETTARE CON I PIC livello base
- PROGETTARE CON I PIC livello avanzato
- PROGRAMMAZIONE IN C per microcontrollori embedded a 8 e 32 bit

POWER ENERGY
- CONTROL TECHNIQUES FOR DC-DC SWITCHING POWER SUPPLIES
- DESIGN OF HIGH-ENERGY-EFFICIENT SWITCHING CONVERTERS
- ELETTRONICA DI POTENZA PER IL FOTOVOLTAICO

FORMAZIONE TECNOIMPRESE

Tecnoimprese è iscritto all'albo enti di formazione della Regione Lombardia

CODICE MIP 2799900

PER MAGGIORI INFORMAZIONI: training@tecnoimprese.it - Tel. 02 210 111 241

Sensori per TUTTI

Se il tuo problema è aggiungere e testare rapidamente un sensore con il tuo prototipo, forse non sapevi che Parallax ne fornisce di convenienti e pronti all'uso: accelerometri, temperatura, umidità, GPS e posizionamento, rotazione, distanza, colore e luce, pressione, movimento, gas, suono, e molti altri.

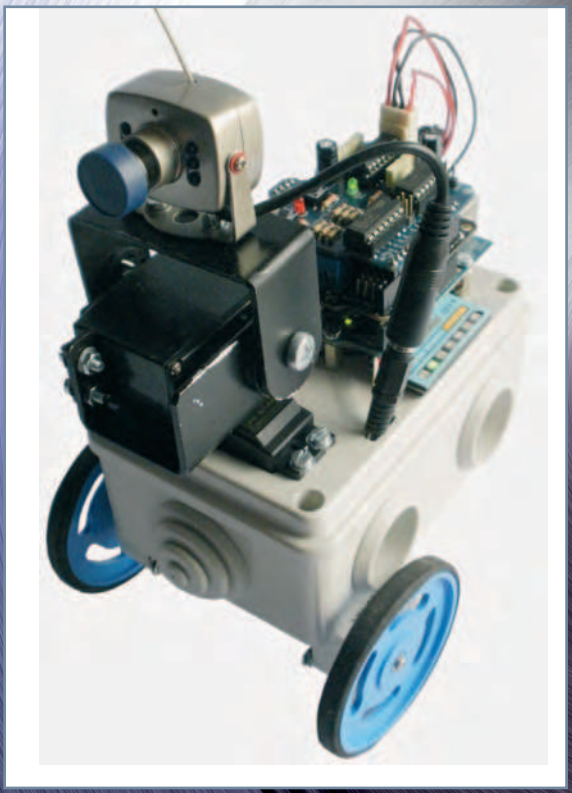
Distribuito da elettroshop.com
tel. 02 66504794 - fax 02 66508225
www.elettroshop.com/sensori

PARALLAX INC
www.parallax.com
Friendly microcontrollers, legendary resources.™

Parallax ed il logo Parallax sono marchi registrati di Parallax Inc.

CODICE MIP 2814118

ROBOTZone



102 ArduinoBOT (parte prima)

UN ROBOT ESPLORATORE CON ARDUINO

I robot esploratore, cui ho attribuito il nome ArduinoBOT, è basato sulla scheda di controllo Arduino (2009 o UNO). Per muoversi, utilizza due ruote azionate da motoriduttori comandati da una scheda motor shield che utilizza un integrato SN754410. Per il comando del robot è previsto l'utilizzo di una coppia di schede Xbee e di un controller del tipo utilizzato per la console giochi Playstation 2. Il robot ha una telecamera wireless mossa da un dispositivo pan & tilt ed è alimentata tramite un convertitore DC/DC. In questo primo articolo vedremo la presentazione del robot, mentre nelle prossime puntate analizzeremo le varie schede che lo compongono, le fasi di costruzione e i programmi di gestione.

di Adriano Gandolfo

ArduinoBOT *(parte prima)*

UN ROBOT *esploratore* con Arduino

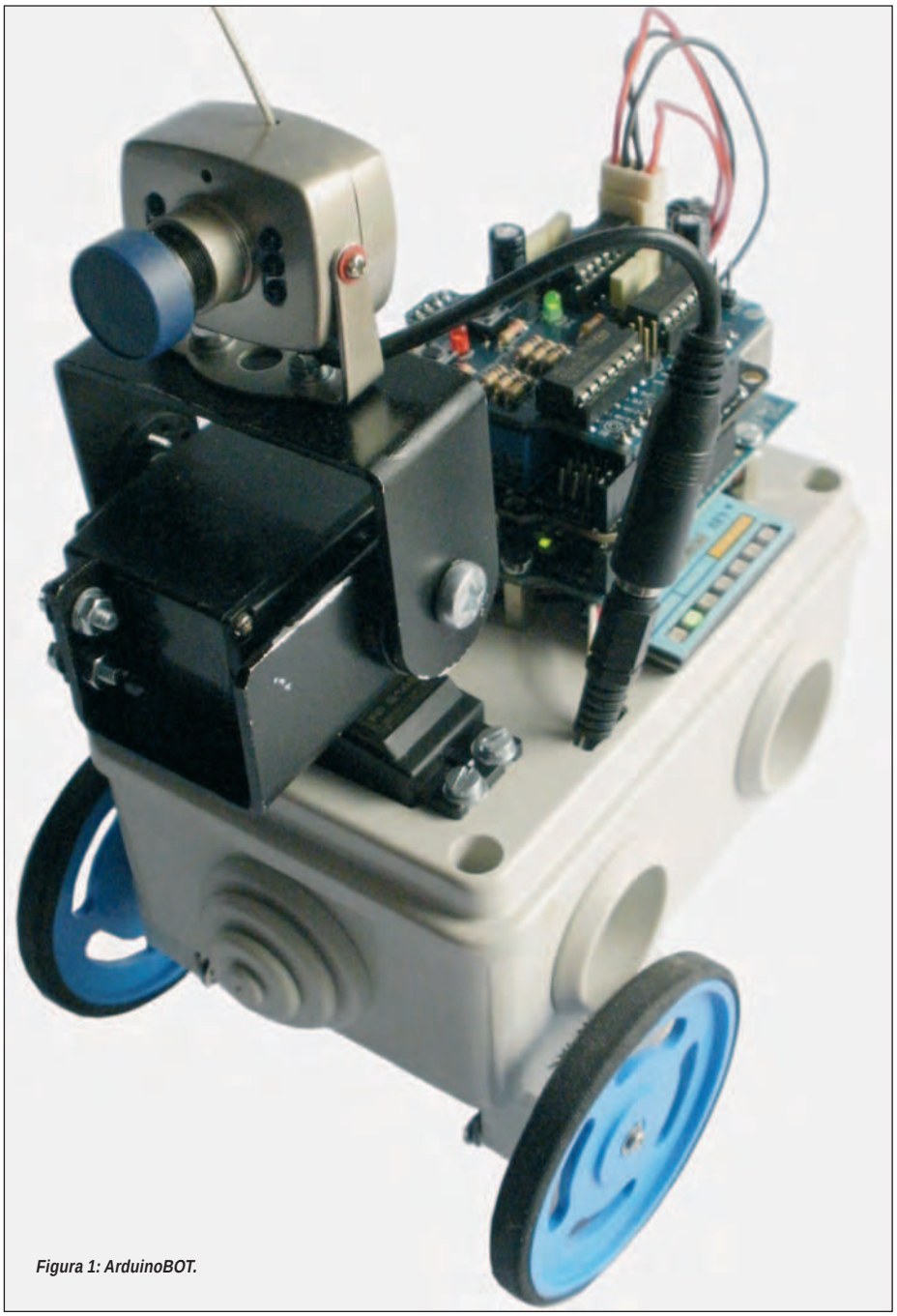


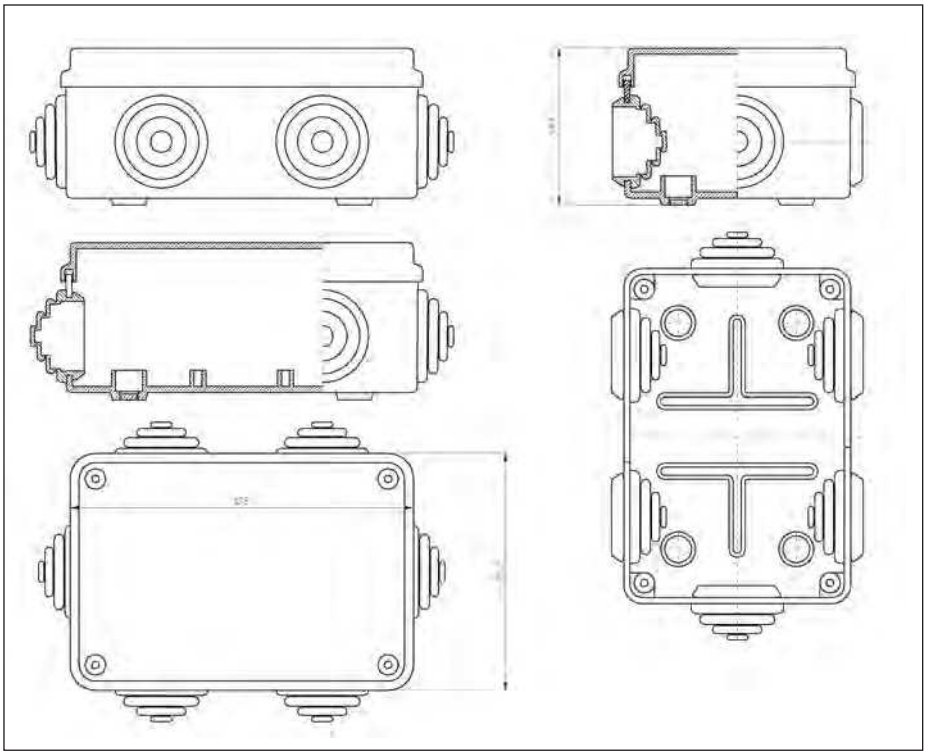
Figura 1: ArduinoBOT.

ArduinoBOT è un robot esploratore basato su una scheda di controllo Arduino. È dotato di due ruote, una telecamera wireless ed è comandato da un controller del tipo utilizzato per la console giochi Playstation 2. In questo primo articolo vedremo la presentazione del robot, mentre nelle prossime puntate analizzeremo le varie schede che lo compongono, le fasi di costruzione e i programmi di gestione

In vendita si possono trovare molti componenti per la realizzazione di piccoli robot. La loro scelta può essere difficoltosa e, una volta assemblata la nostra opera, occorre programmare il processore per comandarla e si può rimanere delusi del risultato. In questa serie di articoli illustreremo come assemblare un robot **(figura 1)** che utilizza come scheda di controllo Arduino, un framework open source basato sul processore ATmega 328. La scheda è stata sviluppata da un team formato da Massimo Banzi, David Cuartielles, Tom Igoe, Gianluca Martino e David Mellis.
Il progetto, sviluppato in Italia, a Ivrea nel 2005, è diventato una base di sviluppo utilizzata in tutto il mondo. Alla scheda sono connesse altre schede di espansione denominate shield, che hanno una piedinatura compatibile con Ar-

Figura 2: modulo di controllo.

Figura 3: telaio base.



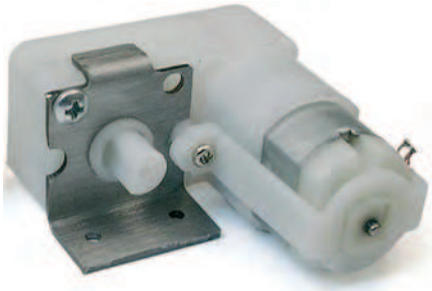


Figura 4: motoriduttore.

Figura 5: una ruota del robot.

Figura 6: ball caster.

duino la quale permette di "impilarle" una sopra l'altra come una torta multistrato. Una scheda permette, tramite un modulo wireless, il dialogo con la stazione di controllo **(figura 2)**; un'altra scheda consente il pilotaggio di due motoriduttori connessi alle ruote che permettono al motore di muoversi; completa il robot un dispositivo pan & tilt che consente il movimento di una piccola telecamera wireless. L'alimentazione del robot è garantita da una batteria da 6 V ricaricabile alloggiata nella base della macchina, la tensione è monitorata tramite un power indicator che tramite l'accensione di una barra di LED ne mostra la carica. Per l'alimentazione della telecamera (8-9 V) è previsto l'utilizzo di un piccolo modulo DC-DC in grado di elevare la tensione di 6 V a quella necessaria.

Il segnale trasmesso dalla telecamera è raccolto da un ricevitore che fornisce in uscita un segnale analogico; le immagini possono essere viste direttamente sullo schermo TV oppure tramite un'interfaccia analogica/digitale sullo schermo del proprio PC dove possono essere eventualmente registrate.

COMPOSIZIONE DEL ROBOT

Passiamo ora ad analizzare le varie parti che formano il robot, per il momento in modo sommario, rimandando alle altre puntate un approfondimento più completo.

Telaio di base

Il telaio del robot è realizzato utilizzando una semplice scatola di derivazione per impianti elettrici **(figura 3)**. In questo caso il modello 685.005 di SCAME. Le dimensioni sono 120 x 80 x 50 cm, di colore grigio. Essendo realizzata in materiale

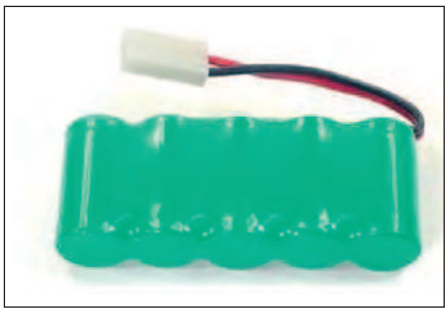


Figura 7: batteria di alimentazione.

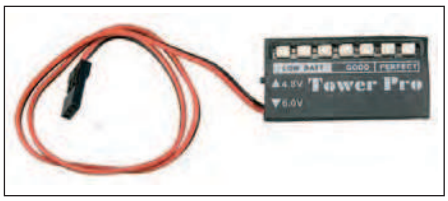


Figura 8: monitor batteria.

plastico è facilmente lavorabile. Su di essa è possibile realizzare le varie aperture per il fissaggio del servo del dispositivo di pan & tilt., l'interruttore di accensione ecc.

Motoriduttori

Il robot è mosso da due motoriduttori **(figura 4)** tipo Gear Motor 2, prodotti da Solarbotics, e ha un rapporto di riduzione di 224:1. Essi sono basati sul disegno originale di Mark Tilden per l'uso nel BIO-Bugs & Robosapien, hanno alberi di uscita doppia e un motore DC a basso consumo di corrente.

Ruote

Le ruote del robot hanno un diametro di 65 mm **(figura 5)**, sono prodotte dalla Solarbotics e realizzate in ABS stampato e si adattano perfettamente all'albero di uscita del motoriduttore Gear Motor 2. Sono completate da speciali elastici per aumentare la trazione e ridurre l'usura.

Supporto posteriore

Il supporto posteriore **(figura 6)** è rappresentato da una ball caster codice 70144 prodotta dalla Tamiya. Nella confezione sono presenti i particolari per realizzare il supporto con un'altezza variabile, compresa tra gli 11 e i 37 mm. Nel nostro robot dovremmo ancora aumentare quest'altezza utilizzando alcuni distanziali metallici.

Batteria di alimentazione

La batteria **(figura 7)** che viene alloggiata nella base del robot è nel tipo Ni-MH dimensione 2/3 A, tensione nominale 6 V, con una capacità minima di 1100 mAh. Per la ricarica della batteria sarà necessario avere un apposito caricabatteria.

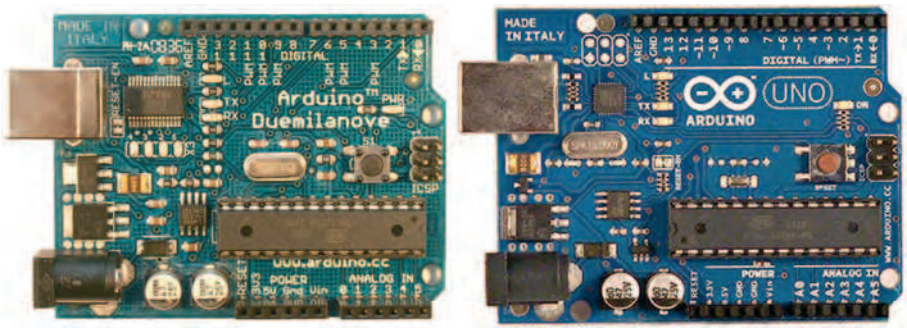


Figura 9: schede Arduino Duemilanove e UNO.

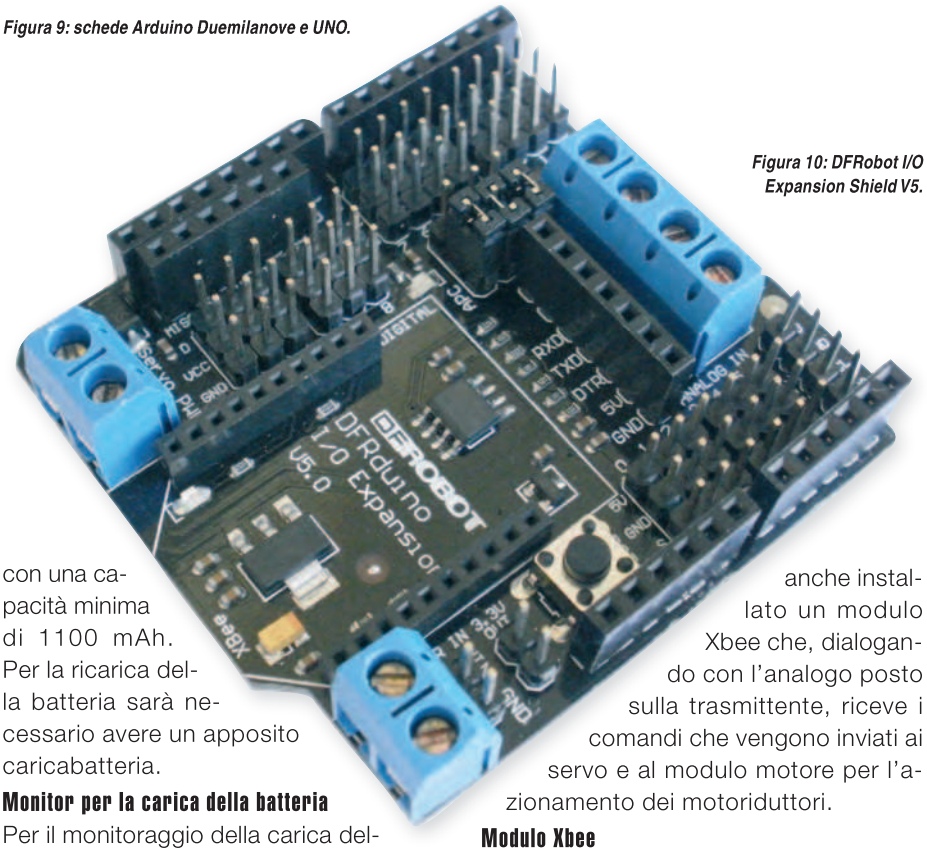


Figura 10: DFRobot I/O Expansion Shield V5.

Monitor per la carica della batteria

Per il monitoraggio della carica della batteria è presente un piccolo circuito **(figura 8)** che, tramite l'accensione di LED, indica quando è il momento di ricaricarla.

Scheda di controllo

La scheda di controllo è una Arduino Duemilanove oppure l'ultima nata, Arduino UNO **(figura 9)**, entrambe basate sul processore ATmega 328. La scheda presenta 14 piedini input/output digitali (di cui 6 utilizzati come uscite PWM), 6 input analogici, un oscillatore di cristallo a 16 MHz, una connessione USB, un ingresso per l'alimentazione.

Scheda controllo servo/comunicazione

Per la connessione dei servo del sistema pan & tilt della telecamera si utilizza una scheda I/O Expansion Shield V5 **(figura 10)** prodotta da DFRobot. Su di essa è anche installato un modulo Xbee che, dialogando con l'analogo posto sulla trasmittente, riceve i comandi che vengono inviati ai servo e al modulo motore per l'azionamento dei motoriduttori.

Modulo Xbee

Sia sul robot, sia sull'unità trasmittente si trovano due moduli Xbee **(figura 11)** prodotti da Digi International Inc. Questi sfruttano il protocollo ZigBee e operano nella banda ISM alla frequenza di 2.4 GHz, implementano una comunicazione seriale come quella presente sulla porta RS232 e permettono di compiere collegamenti wireless.

Scheda motore

La scheda motore **(figura 12)** è basata sullo stampato Motor Control v3.0. La scheda deve essere montata seguendo le semplici istruzioni che si trovano anche in Rete. Su di essa è montato un integrato SN754410 compatibile pin to pin con l'integrato L293D, che è un quadruplo ponte H per il pilotaggio di due motori CC. I segnali di comando vengono ricevuti

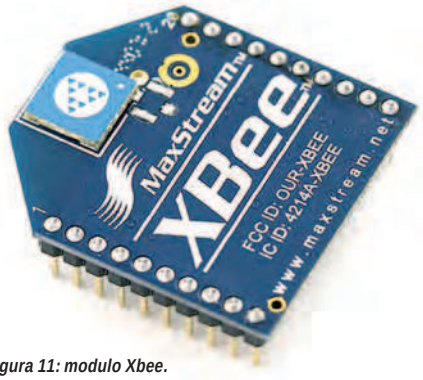


Figura 11: modulo Xbee.

tramite il modulo Xbee presente sulla scheda I/O Expansion Shield.

Dispositivo pan & tilt

Il dispositivo pan & tilt **(figura 13)** permette la movimentazione della telecamera nelle quattro direzioni: destra-sinistra (pan) e alto-basso (tilt). Il modello montato sul robot è stato autocostruito, ma si trovano in commercio vari modelli come per esempio il tipo BPT-KT prodotto dalla Lynxmotion, nel cui kit sono già compresi i due servomotori.

Convertitore DC/DC

Per elevare la tensione della batteria si utilizza un mini Convertitore DC/DC prodotto da Pololu, codice 791 **(figura 14)**. Dalle dimensioni particolarmente ridotte, è in grado di convertire una tensione continua compresa tra 1,5 e 16 volt in una tensione di uscita compresa tra 2,5 e 9,5 volt (regolabile tramite trimmer).

Telecamera wireless

Per riprendere quello che vede il nostro robot si utilizzerà una piccola telecamera a colori di tipo CMOS **(figura 15)**. Questa è dotata di un trasmettitore sulla frequenza di 1,2 MHz. Di solito viene venduta con il relativo ricevitore che permette di vedere quanto ripreso. L'alimentazione è fornita dalla batteria del robot elevata tramite in convertitore DC/DC. Il movimento della telecamera sarà garantito dal dispositivo pan & tilt che sarà azionato tramite il controller connesso alla stazione di controllo. La presenza di LED infrarossi permette la ripresa anche in presenza di scarsa illuminazione.

STAZIONE DI CONTROLLO

La stazione di controllo del robot utilizza un'altra scheda Arduino su cui è montato un piccolo shield **(figure 2** e **16)** che permette l'interfacciamento di un modu-

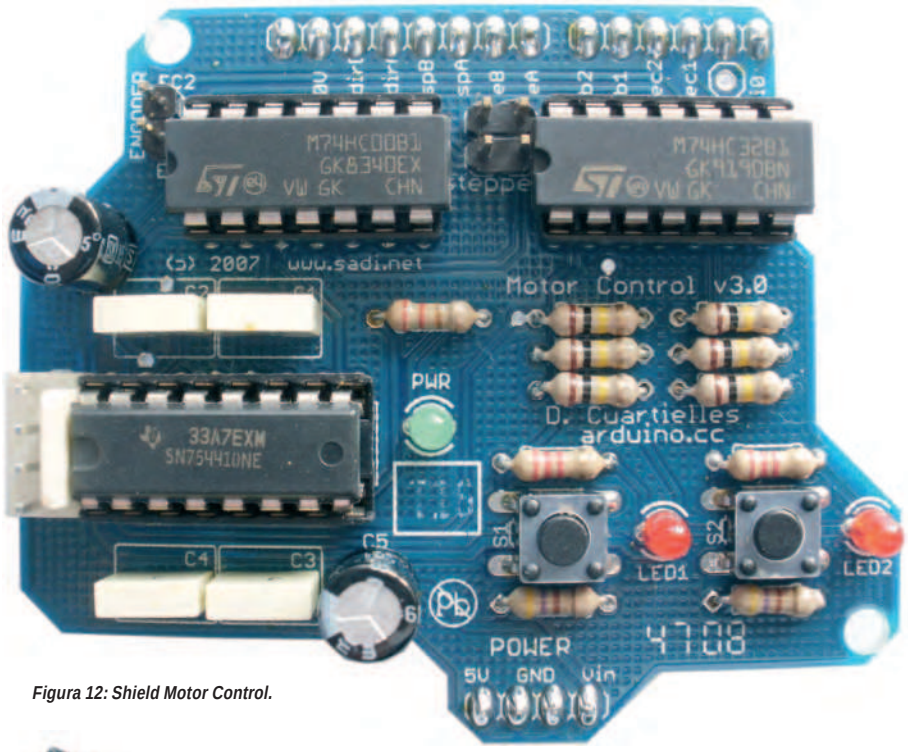


Figura 12: Shield Motor Control.

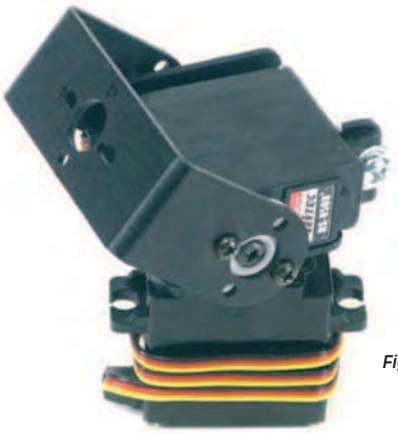


Figura 13: dispositivo pan & tilt.

Figura 14: convertitore DC/DC.

CODICE MIP 2797072

lo Xbee **(figura 11)** e di un controller per console Playstation 2. Tramite quest'ultimo è possibile comandare il movimento sia del robot sia della telecamera. In tal modo non è necessario essere direttamente connessi a un PC per il controllo del robot. Formano questa stazione di controllo i seguenti elementi:

Figura 15: telecamera wireless.

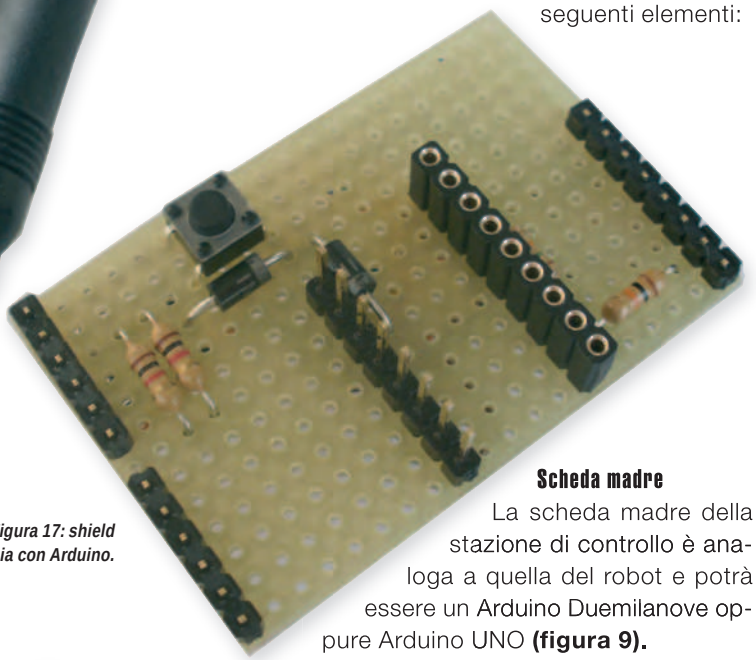


Figura 17: shield d'interfaccia con Arduino.

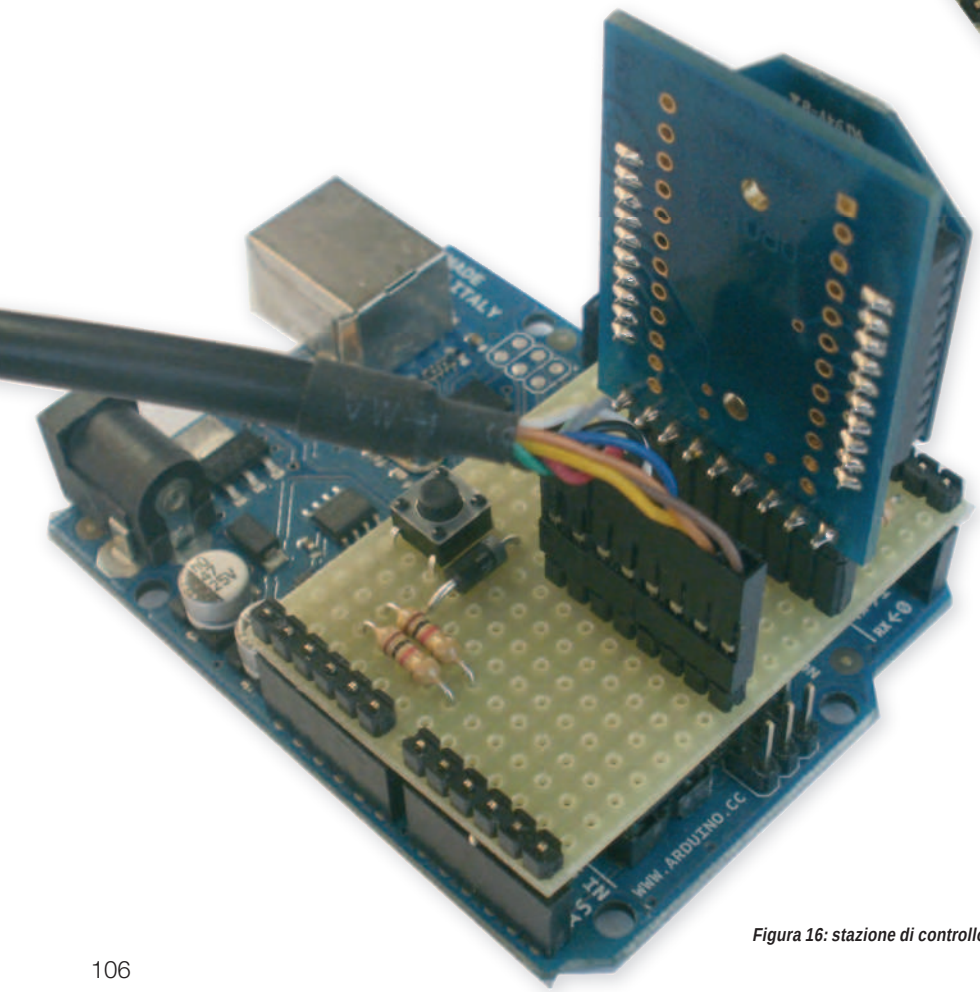


Figura 16: stazione di controllo.

Scheda madre

La scheda madre della stazione di controllo è analoga a quella del robot e potrà essere un Arduino Duemilanove oppure Arduino UNO **(figura 9).**

Shield d'interfaccia

Per interfacciare il modulo Xbee e il controller con la scheda Arduino è previsto l'utilizzo di un piccolo shield da autocostruirsi **(figura 17).** Su di esso, oltre ai vari connettori d'interfaccia, sono presenti un pulsante per il reset e alcuni componenti passivi.

Scheda di interfaccia per modulo XBee

La scheda consente l'istallazione del modulo Xbee **(figura 18)** e su di essa è presente un regolatore di tensione, poiché l'alimentazione del modulo è di 3,3 V. Sono inoltre presenti due LED che consentono il costante monitoraggio dell'attività della scheda stessa. Il modello scelto è XBee - Simple Board prodotta da Droids.

Controller per console per Playstation 2

Per impartire i comandi al nostro robot si utilizza un controller utilizzato nella console per videogiochi Playstation 2 di Sony **(figura 19)**. Esso dispone di sedici pulsanti di cui quattro formano un pad direzionale, quattro tasti (identificati dai simboli triangolo, cerchio, croce e quadrato) e due joystick analogici. Il controller è connesso alla scheda Arduino che ne decodifica i segnali.

PER *approfondire...*

- Sito del produttore della scheda Arduino, dove trovare informazioni sulla scheda e scaricare l'ambiente di sviluppo: http://www.arduino.cc/
- Informazioni sulla scheda controllo motori: http://arduino.cc/playground/MotorControlShieldV3/0
- Sito del produttore della I/O Expansion Shield: http://www.dfrobot.com/
- Sito del produttore dei motoriduttori: http://www.solarbotics.com/
- Sito di Lynxmotion per il dispositivo pan & tilt: http://www.lynxmotion.com
- Sito del produttore del modulo DC/DC: http://www.pololu.com/
- Sito del produttore dei moduli Xbee: http://www.digi.com/
- Sito per la scheda XBee - Simple Board: http://www.droids.it

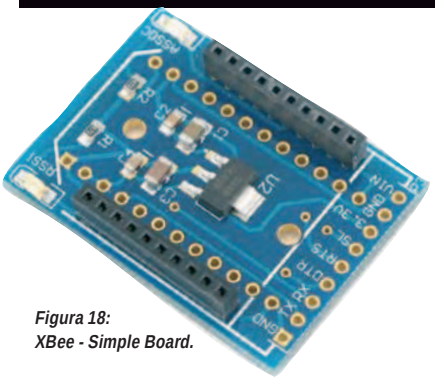


Figura 18: XBee - Simple Board.

Figura 19: controller per console per Playstation 2.

Cavo interfaccia

Il controller ha uno speciale connettore per il collegamento alla console di gioco. Per evitare di dover tagliare il cavo è possibile utilizzare un apposito adattatore **(figura 20)** reperibile sul sito Lynxmotion con il codice PS2C-01.

RICEZIONE DEL SEGNALE VIDEO

Per la ricezione del segnale video potremo utilizzare un normale televisore il cui ingresso analogico andrà connesso all'uscita del ricevitore presente nella confezione della telecamera. Altra possibilità è interfacciare il ricevitore con un PC o un computer portatile **(figura 21)** tramite un apposito convertitore che convertirà il segnale analogico in digitale. In questo modo le immagini potranno eventualmente essere registrate.

Figura 20: adattatore per il controller Playstation 2.

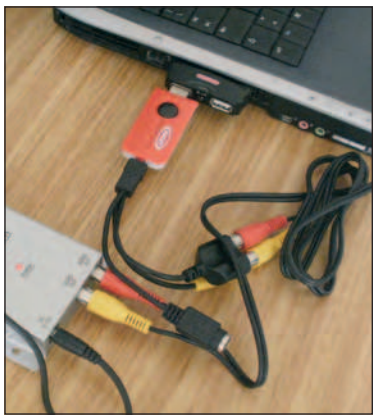


Figura 21: collegamento tra ricevitore AV e PC.

L'AMBIENTE DI SVILUPPO

Per la redazione dei programmi necessari al funzionamento del robot si utilizzerà l'ambiente di sviluppo Arduino che contiene un editor di testo per scrivere il codice, una zona messaggio, una console di testo, una barra degli strumenti con pulsanti per le funzioni comuni e una serie di menu. Questo permette di collegarsi all'hardware Arduino per caricare i programmi e comunicare con esso. I programmi scritti per Arduino si chiamano sketch. Questi sketch sono scritti nell'editor di testo, cosa che permette operazioni di taglia/incolla e di ricerca/sostituzione di testo. L'area messaggi fornisce un feedback durante il salvataggio, l'esportazione e anche la segnalazione degli errori. I pulsanti della barra degli strumenti consentono di verificare e aggiornare programmi, creare, aprire, salvare e utilizzare il serial monitor.

CONCLUSIONI

In questa prima parte abbiamo illustrato rapidamente i principali componenti che formano il nostro robot. Nelle prossime puntate scenderemo più in dettaglio analizzando le varie schede, i componenti e inizieremo con le fasi di montaggio. ❑

Potete vedere alcuni filmati preliminari del robot pubblicati su YouTube:
http://youtu.be/unt1Hct5rSA
http://youtu.be/wO20bwAAkk8

CODICE MIP 2805079

Tele-gestione della caldaia col gsm

Tutto via cellulare:
Termostato +
Combinatore +
Apri-cancello ...
Con carica batterie
litio e relè integrati

www.carrideo.it/moduli_embedded.htm
www.carrideo.it/telecaldaia.htm

CODICE MIP 2803041

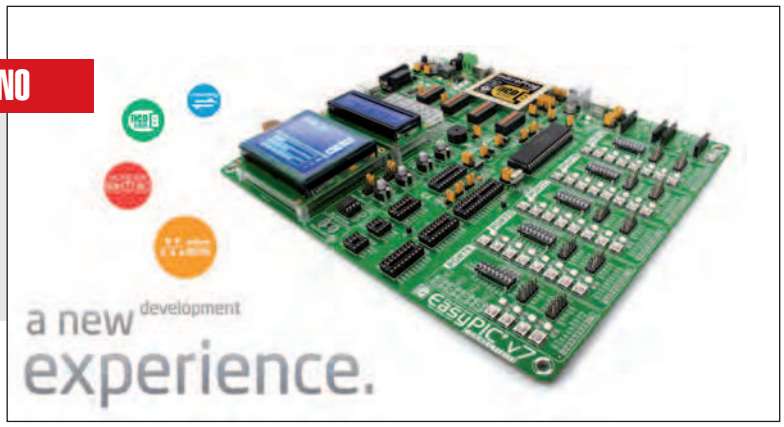
Lo shop dei prodotti Inware Edizioni...

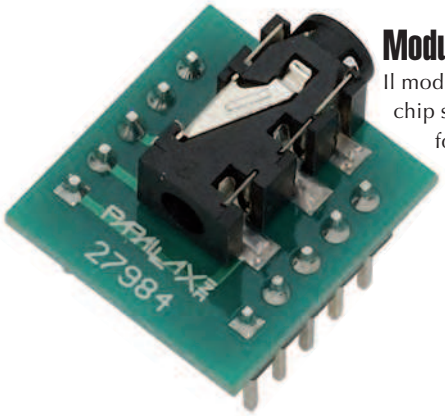
PRIMO PIANO

La nuova scheda EasyPIC7

EasyPIC v7 è la settima generazione della famosa scheda di sviluppo PIC. E' il prodotto di conoscenze accumulate nell'arco di 10 anni e rappresenta lo stato dell'arte nella progettazione, funzionalità e qualità. E' amata dai neofiti e hobbisti così come dai professionisti e utilizzata nell'insegnamento e sviluppo da migliaia di persone. La scheda viene fornita con un PIC18F45K22.

CODICE: EASYPIC7 PREZZO: € 175,45





Modulo di ricezione Radio FM

NOVITA'

Il modulo di ricezione radio FM utilizza un chip stereo radio tuner FM RDA5807SS che fornisce una soluzione semplice al vostro microcontrollore per ricevere le stazioni radio FM locali. Usando l'antenna onboard e il jack per auricolari/cuffie, potete facilmente creare la vostra radio portabile!

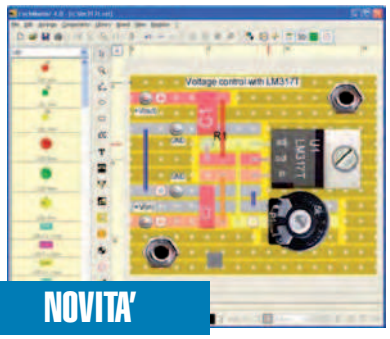
CODICE: 27984 PREZZO: € 18,76

NOVITA'

EasyGPS Board

La scheda EasyGPS è una soluzione compatta per aggiungere la funzionalità GPS al vostro dispositivo. Fornisce LEA-6S e high performance u-blox 6 positioning engine. La scheda è progettata nel formato mikroBUS. La comunicazione è effettuata utilizzando UART o USB. La scheda può essere usata sia con un microcontrollore oppure i dati possono essere acquisiti usando una applicazione PC attraverso l'USB. Tutte le linee di comunicazione devono essere guidate con tensione logica a 3.3V.

CODICE: EASYGPS BOARD PREZZO: € 47,19



NOVITA'

LochMaster

Uno strumento di sviluppo per la progettazione di schede. Possiede utili funzionalità per progettare, documentare e testare una scheda. Si trovano caratteristiche quali autogenerazione di liste di componenti, test di connessione, una libreria customizzabile con un vasto numero di simboli e componenti, e molto altro. Nella nuova versione, il software è molto più realistico. La nuova libreria estesa riprende quasi perfettamente i componenti così come sono nella realtà. Un'altra ottima caratteristica è l'editor di schede integrato. E' ora possibile creare le proprie schede, se la scheda desiderata non è tra quelle già definite.

CODICE: LOCHMASTER PREZZO: € 48,28

Ebook sui display LCD

NOVITA'

Un testo digitale in formato PDF ad alta risoluzione sull'uso dei display LCD alfanumerici basati su controller HD44780 o compatibile. Una guida essenziale per conoscerli ed utilizzarli sia cn i microcontrollori sia con il PC attraverso la porta seriale e parallela.

CODICE: EB-LCD PREZZO: € 14,52

Modulo Bluetooth

OFFERTA

F2M03C2 è un modulo Bluetooth SMD operante in classe 2 adatto per comunicazioni dati e voce, si interfaccia via UART o USB e consente la trasmissione dati fino al massimo datarate consentito da Bluetooth (723.2Kbps). Il dispositivo è dotato di una interfaccia audio digitale 13-bit PCM bidirezionale (campionamento a 8kHz) oltre a vari I/O analogici e digitali ed un'interfaccia I2C.

CODICE: F2M03C2 PREZZO: ~~€ 66.55~~ € 39,69

GPS Facile

OFFERTA

Connetti il GPS (Sistema di Posizionamento Globale) al tuo PC o alla tua scheda di sviluppo con semplicità con la board SmartGPS. Basata sul modulo on-board LEA-5S GPS/GALILEO ad alte prestazioni basato su u-blox, supporta antenne attive e passive.

CODICE: EP-EASYGPS PREZZO: ~~€ 59,29~~ € 47,19

7 eBook ad un prezzo eccezionale

BUNDLE

A meno di trenta euro ben 7 ebook tecnici. Il bundle comprende i seguenti titoli: Smartcard, PICmicro, CPLD, MikroC, Amplificatori operazionali, Alimentatori Switching, Display LCD.

CODICE: BUNDLE EBOOK PREZZO: € 34,50

BEST SELLER

Una collezione di circuiti

Volete costruirvi un cavo adattatore particolare? Volete uno schema per la sezione di alimentazione del vostro progetto? In CIRCUITS PACK troverete senz'altro la soluzione che fa per voi. Oltre 250 schemi elettrici e progetti pronti da realizzare.

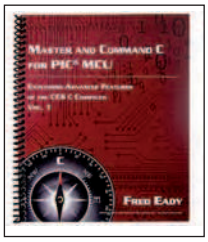
CODICE: CIRCUITS PACK PREZZO: € 6,99

Lo shop dei prodotti Inware Edizioni... ma non solo!

Usa IEmobile! Abbonamento CLUB Firmware digitale al prezzo del premium!"

Usa IEMOBILE!

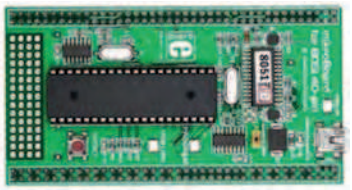
Sottoscrivi un abbonamento CLUB al prezzo del PREMIUM! L'abbonamento CLUB prevede 6 numeri di Firmware Edizione Digitale, compreso 1 anno di iscrizione al Club di Firmware e 1 anno di Fare Elettronica



Master and Command C for PIC®MCU

Master and Command C for PIC®MCU fornisce ai lettori l'opportunità di scrutare il lavoro di Fred (l'autore di questo libro) mentre esplora alcune tra le più avanzate funzioni built-in del compilatore CCS. I neofiti così come gli utenti più esperti impareranno molti trucchi e scorciatoie per migliorare il proprio stile di programmazione.

CODICE: CCS-BOOK2 PREZZO: € 14,52



mikroBoard per 8051

mikroBoard for 8051 40-pin with AT89S8253 fornisce un programmatore integrato e debugger in-circuit, regolatore di alimentazione e circuiti di reset, così come un'area per prototipazione. Progettata per essere usata nell'ambiente di sviluppo multi-processore UNI-DS6, può anche essere utilizzata come processor board stand-alone.

CODICE: UNI-8051 PREZZO: € 54,33

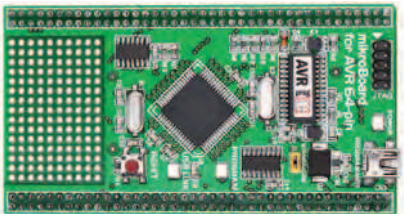
mikroBoard for dsPIC with dsPIC30F6014A

mikroBoard for dsPIC with dsPIC30F6014A fornisce un programmatore integrato e debugger in-circuit, regolatore di alimentazione e circuiti di reset, così come un'area per prototipazione. Progettata per essere usata nell'ambiente di sviluppo multi-processore UNI-DS6, può anche essere utilizzata come processor board stand-alone.

CODICE: UNI-DSPIC PREZZO: € 54,33

mikroBoard for PIC 40-pin con PIC16F887

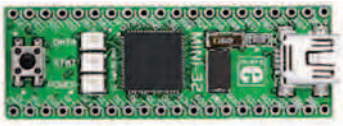
mikroBoard for PIC 40-pin con PIC16F887 fornisce un programmatore integrato e debugger in-circuit, regolatore di alimentazione e circuiti di reset, così come un'area per prototipazione. Progettata per essere usata nell'ambiente di sviluppo multi-processore UNI-DS6, può anche essere utilizzata come processor board stand-alone.



CODICE: UNI-PIC-40 PREZZO: € 54,33

MINI-32 Board

MINI-32 è una piccola scheda di sviluppo contenente il microcontrollore PIC32MX534F064H. Opera con una alimentazione di 3.3V. Il regolatore di voltaggio on-board permette alla scheda di essere alimentata direttamente da cavo USB. E' compatibile a livello pin con i microcontrollori PIC16F887 e PIC18(L)F45K20 e si adatta perfettamente all'interno del socket standard DIP40. La scheda è equipaggiata con un oscillatore a cristallo SMD e con un cristallo a 32.768KHz che può essere usato per il modulo interno RTCC. Possiede un pulsante di reset e tre LED di segnale. La scheda è pre-programmata col veloce bootloader HID USB, in modo tale che non sia necessario programmatori esterni per lo sviluppo.



CODICE: MINI-32 BOARD PREZZO: € 27,83

prossimamente su Fare Elettronica

COMING...

APPLICAZIONI CON IL CUBLOC CUSB22R

Ecco un bellissimo esempio di applicazione del CUSB22R per riempire un contenitore di liquido a livelli prestabiliti. Il programma viene scritto sia in Basic che Ladder Logic.

ESPERIMENTI CON GLI OSCILLATORI

Il funzionamento degli oscillatori e alcune semplici implementazioni pratiche che potrete autonomamente riprodurre disponendo di una semplice bread-board, un paio di pile da 9V e pochi altri componenti.

CORSO MIKROBASIC: ANTIRIMBALZO AVANZATO

In questo articolo tratteremo un tema sempre al centro dela progettazione elettronica: l'eliminazione del rimbalzo elettrico. Un problema che affligge gli sviluppatori e I progettisti ma che si può eliminare del tutto utilizzando diverse tecniche, sia software che hardware.

COME RISPARMIARE?

Ordinare direttamente tramite internet conviene! Potrai infatti aggiudicarti dei buoni sconto (coupon) con le modalità:

COUPON ABBONAMENTO ONLINE FE E FW
Sottoscrivendo un nuovo abbonamento o rinnovando il proprio online, direttamente tramite IE SHOP, si ha diritto ad un coupon del 15%! **Qualora ci si abbona (sempre on-line) a Fare Elettronica e Firmware insieme il coupon è del 20%!**

COUPON QUIZ "ELETTROQUIZ"
Rispondendo ai Quiz presente ogni mese su Fare Elettronica si avrà diritto ad un coupon del 10%. Tutte le risposte pervenute saranno accuratamente valutate dagli autori dei quesiti e, in caso di risposta esatta (oltre al Coupon) potrai vincere fantastici premi!

COUPON "ACQUISTI PREMIATI"
I tuoi acquisti su www.ieshop.it vengono premiati con un coupon del 10% se il tuo ordine supera i 100 EUR (IVA e spese di trasporto escluse). Ma se il tuo ordine supera i 200 EUR, lo sconto sale al 20%!

IMPORTANTE!
- Il coupon può essere speso **esclusivamente** ordinando sul sito www.ieshop.it
- Il coupon non potrà essere utilizzato in caso di nuova sottoscrizione o rinnovo dell'abbonamento, è invece utilizzabile per tutti gli altri prodotti presenti su www.ieshop.it
- Ogni coupon ha una scadenza, non dimenticarlo!
- Lo slogan "più acquisti più risparmi" pecca senz'altro di originalità, ma in questo caso è molto azzeccato: i coupon sono infatti utilizzabili solo una volta, usalo bene!
- I coupon non sono cumulabili, e vanno utilizzati singolarmente, secondo il criterio cronologico di assegnazione.

CONTI CORRENTI POSTALI - Ricevuta di Versamento BancoPosta

sul C/C n. 70107552 di Euro ,

importo in lettere

INTESTATO A:
INWARE EDIZIONI SRL

ESEGUITO DA:

VIA - PIAZZA

CAP

LOCALITÀ

BOLLO DELL'UFF. POSTALE

AVVERTENZE
Il Bollettino deve essere compilato in ogni sua parte (con inchiostro nero o blu) e non deve recare abrasioni, correzioni o cancellature.
La causale è obbligatoria per i versamenti a favore delle Pubbliche Amministrazioni.
Le informazioni richieste vanno riportate in modo identico in ciascuna delle parti di cui si compone il bollettino.

CAUSALE

CONTI CORRENTI POSTALI - Ricevuta di Accredito BancoPosta

sul C/C n. 70107552 di Euro ,

TD 451

importo in lettere

INTESTATO A:
INWARE EDIZIONI SRL

CAUSALE

ESEGUITO DA:

VIA - PIAZZA

CAP LOCALITÀ

BOLLO DELL'UFF. POSTALE
codice bancoposta

IMPORTANTE: NON SCRIVERE NELLA ZONA SOTTOSTANTE
importo in euro numero conto td

AUT. DB/SISB/E 20017 DEL 11.04.2006

70107552 < 451>

IMPORTANTE: NON SCRIVERE NELLA ZONA SOPRASTANTE!

AVVERTENZE

Il Bollettino deve essere compilato in ogni sua parte (con inchiostro nero o blu) e non deve recare abrasioni, correzioni o cancellature.

La causale è obbligatoria per i versamenti a favore delle Pubbliche Amministrazioni.

Le informazioni richieste vanno riportate in modo identico in ciascuna delle parti di cui si compone il bollettino.

INDAGINE SUI LETTORI

Aiutaci a conoscerti meglio!
Con il tuo aiuto riusciremo ad offri orti una rivista sempre più in linea con le tue aspettative..

COMPILA ONLINE
IL QUESTIONARIO ALL'INDIRIZZO
WWW.FARELETTRONICA.COM/SURVEY

Per ringraziarti per il tuo tempo e la tua cortesia, ti invieremo gratuitamente un bellissimo eBook del valore di **14,52 EUR**!

PRODOTTI

Se vuoi ordinare "offline" ovvero senza usare internet, usa il seguente modulo d'ordine **PRODOTTI**

COD. PRODOTTO	DESCRIZIONE	PREZZO UNITARIO	Q.tà	Totale
			SUB TOTALE	
			SPESE DI SPEDIZIONE	
			TOTALE	

Nota. Se le righe non sono sufficienti, utilizzare più copie del modulo.
Per maggiori dettagli sulle spese di spedizione e i metodi di pagamento vedi retro cartolina.

METODI DI PAGAMENTO

☐ **BONIFICO BANCARIO** ☐ **BOLLETTINO POSTALE** ☐ **CONTRASSEGNO**

FE 317

☐ MasterCard ☐ VISA ☐ postepay

TITOLARE. . . . . .
NUMERO . . . . . .
DATA DI SCADENZA. . . . . CODICE DI SICUREZZA . . . . . .

INVIARE IL MODULO D'ORDINE A: Inware Edizioni srl, Via Giotto,7 - 20032 Cormano (MI) - oppure via FAX allo 02 66508225
(Inviando questo modulo si accettano i termini e le condizioni riportate a tergo)

Nome . . . . . Cognome . . . . . .
Indirizzo . . . . . n° . . . . . .
Cap . . . . . Città . . . . . Prov . . . . . .
Tel . . . . . Fax . . . . . Email . . . . . .
Ragione Sociale . . . . . .
Codice fiscale . . . . . Partita IVA . . . . . .

DATA . . . . . FIRMA . . . . . ☐ Fattura

ABBONAMENTO

Sì, mi abbono a Fare Elettronica

FE 317

☐ Abbonamento TRIAL a 3 numeri di FARE ELETTRONICA a soli € 16,50 anzichè € 18,00
☐ Abbonamento PREMIUM a 11 numeri di FARE ELETTRONICA a soli € 55,00 anzichè € 66,00
☐ Abbonamento PRO a 11 numeri di FARE ELETTRONICA include il Cd dell'annata 2010 a soli € 59,50 anzichè € 96,00

METODI DI PAGAMENTO (per maggiori dettagli vedi retro cartolina)

☐ **BONIFICO BANCARIO** ☐ **BOLLETTINO POSTALE** ☐ **ALLEGO ASSEGNO** (Intestato a Inware Edizioni)

☐ MasterCard ☐ VISA ☐ postepay

PayPal fare il pagamento su paypal@inwaredizioni.it

TITOLARE. . . . . .
NUMERO . . . . . .
DATA DI SCADENZA. . . . . CODICE DI SICUREZZA . . . . . .

INVIARE IL MODULO D'ORDINE A: Inware Edizioni srl, Via Giotto,7 - 20032 Cormano (MI) - oppure via FAX allo 02 66508225
(Inviando questo modulo si accettano i termini e le condizioni riportate a tergo)

Nome . . . . . Cognome . . . . . .
Indirizzo . . . . . n° . . . . . .
Cap . . . . . Città . . . . . Prov . . . . . .
Tel . . . . . Fax . . . . . Email . . . . . .
Ragione Sociale . . . . . .
Codice fiscale . . . . . Partita IVA . . . . . .

DATA . . . . . FIRMA . . . . . ☐ Fattura

ISTRUZIONI PER L'ORDINE E SPESE DI SPEDIZIONE

Il modulo d'ordine dovrà essere compilato in tutte le sue parti ed inviato via posta o via fax ai recapiti indicati sul modulo d'ordine stesso. Gli ordini potranno essere fatti anche direttamente online dal sito www.ieshop.it. In questo caso non sarà necessario inviare il modulo d'ordine. Le spese di spedizione ammontano a euro 8,50 a cui vanno aggiunti euro 3,50 se si sceglie di pagare in contrassegno.

METODI DI PAGAMENTO

Si accettano pagamenti ino in contrassegno, carta di credit, bollettino postale o bonifico all'ordine. Per il contrassegno verrà applicata una spesa aggiuntiva di euro 3,50 per le spese di contrassegno. Forme diverse di pagamento devono essere previamente concordate.

Bonifico bancario

Appoggiato su Poste Italiane IBAN: IT 68 I 07601 01600 000070107552 intestato a Inware Edizioni srl.

Bollettino postale

Versamento sul conto corrente postale n. 70107552 intestato a Inware Edizioni srl. Inviare la ricevuta (o copia) del versamento.

Contrassegno

La merce verrà pagata direttamente al corriere alla consegna della merce. Il pagamento in contrassegno comporta l'addebito di euro 3,50per spese di contrassegno.

Carta di Credito

Il pagamento con carta di credito può essere effettuato specificando i dati della carta via posta, email, web, fax o telefono. Per gli ordini effettuati sul sito il pagamento mediante carta di credito verrà effettuato attraverso una connessione sicura SSL per garantire la massima sicurezza.

TERMINI E CONDIZIONI

Tutti i prodotti sono coperti da garanzia. La garanzia copre tutti i difetti di fabbricazione entro un anno dal ricevimento della merce. Tutti i prodotti non funzionanti per uso improprio o incuria non saranno ritenuti in garanzia e saranno addebitati gli eventuali costi di riparazione. Tutti i prodotti verranno riparati e/o sostituiti direttamente dal produttore. Non sono coperti da garanzia i componenti elettronici (microprocessori, memorie, ecc.) La garanzia dei prodotti si intende F.co ns. sede, le eventuali spese di trasporto sono a carico del cliente salvo accordi diversi. Per dar corso alla riparazione/sostituzione in garanzia è necessario seguire l'apposita procedura di RMA.

PRIVACY

Ai sensi del Decr. Lgs. 196/2003 la informiamo che i dati trasmessi verranno impiegati coi principali scopi di indagini di mercato e nelle modalità previste dallo stesso, prevalentemente con mezzi informatici. Il conferimento, di norma facoltativo, è obbligatorio per permettere il rapporto commerciale. È in ogni caso fatto diritto dell'interessato esercitare i propri diritti, nei modi previsti dal "Titolo II art. 7" della legge sopra citata, scrivendo a Inware Edizioni srl, Via Giotto 7 – 20032 Cormano o tramite email all'indirizzo info@inwaredizioni.it

---

ABBONARSI ALLE RIVISTE INWARE EDIZIONI CONVIENE!

I vantaggi per gli abbonati sono, oltre al prezzo bloccato per un anno, la ricezione del numero direttamente a casa con la garanzia di ricevere tutti i numeri. Inoltre un vistoso risparmio che, nel caso dell'abbonamento PLUS, ammonta a ben 36,50 euro. L'abbonamento ha una durata di 12 mesi e comporta l'invio di 11 numeri di Fare Elettronica. Eventuali variazioni di indirizzo andranno comunicate tempestivamente alla Redazione che provvederà a registrare il cambiamento senza alcuna spesa aggiuntiva. L'abbonamento decorrerà dal primo numero raggiungibile alla data di avvenuto pagamento. Non sono previsti rimborsi in caso di disdetta dell'abbonamento.

METODI DI PAGAMENTO

Bonifico bancario

Appoggiato su Poste Italiane IBAN: IT 68 I 07601 01600 000070107552 intestato a Inware Edizioni srl.

Bollettino postale

Versamento sul conto corrente postale n. 70107552 intestato a Inware Edizioni srl. Inviare la ricevuta (o copia) del versamento indicando nella causale: "Abbonamento Fare Elettronica"

Carta di Credito

Il pagamento con carta di credito può essere effettuato specificando i dati della carta via posta, email, web, fax o telefono. Per gli ordini effettuati sul sito il pagamento mediante carta di credito verrà effettuato attraverso una connessione sicura SSL per garantire la massima sicurezza.

Assegno bancario

E' possibile spedire un assegno bancario insieme a questo coupon.

Paypal

Inviare il pagamento a paypal@invaredizioni.it.

LA RIVISTA DI ELETTRONICA PIÙ LETTA AL MONDO (OLTRE 400MILA COPIE IN 12 LINGUE) DA OGGI IN ITALIANO!

*luglio/agosto numero doppio

Creata da veri appassionati di elettronica con la produzione di molti progetti pratici, tutti originali e sempre innovativi.

Progetti di elevata qualità ideali per l'autocostruzione nei campi dei microcontrollori, audio high-end, RC modelling, RF e test & measurement.

Gli sviluppi sulle nuove tecnologie, grazie ad una serie di articoli mensili, sempre pratici e visti da angolazioni diverse.

L'unica rivista al mondo che può vantare laboratori di progettazione e di disegno dei PCB, con la produzione e vendita di kit.

Il nuovo numero ti aspetta in edicola a partire dal 20 del mese precedente*

www.elektor.it

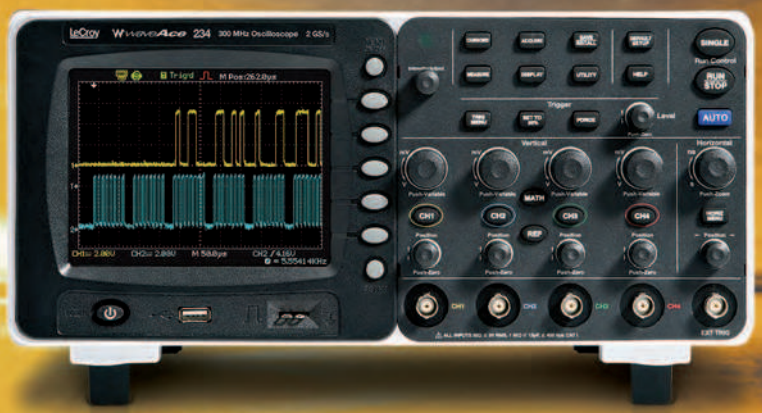
CODICE MIP 2314078

RS

LeCroy

Educate i vostri studenti a pretendere il meglio risparmiando!

Oscilloscopi WaveAce - Da 40 MHz fino a 300 MHz



LeCroy, la qualità che non compromette il vostro budget.

Ora disponibile!

acquistando un oscilloscopio WaveAce riceverai gratis il nuovo EduPack Advanced, software educazionale sviluppato da LeCroy che consente il facile apprendimento delle teorie e tecniche basilari dell'analisi dei segnali elettronici e assiste sia lo studente che il docente nelle attività di apprendimento o di insegnamento.

waveAce EduPack ADVANCED

30% di sconto su WaveAce 102 !!

~~€830~~ **€570**

CODICE MIP 2811547

RS

www.rs-components.it